JEDE AUSGABE MIT FIRMENREGISTER UND PRAKTIKUMSBÖRSE

04/2009 € 6,90

making games MAGAZIN

MAGAZIN FÜR SPIELE-ENTWICKLUNG UND BUSINESS-DEVELOPMENT

DIGITALE DISTRIBUTION

GROSSER ERFOLG MIT KLEINEN MITTELN: TIPPS VON TELLTALE-CHEF DAN CONNORS, AKTUELLE FALLSTUDIEN UND ERFAHRUNGSWERTE

PUBLISHER-GRÜNDUNG

SO ENTSTEHT DAS ERSTE CHARITY-PROJEKT DER GAMESBRANCHE

DS-ENTWICKLUNG

WIE SIE FRAGMENTIERUNGS- UND LEAKPROBLEME UMGEHEN

EVE ONLINE

GEHEIMNISSE EINER REIBUNGS-LOSEN ADDON-AUSLIEFERUNG

© CAPCOM., LTD. ALLE RECHTE VORBEHALTEN

So erweckt man Monster zum Leben

Capcom, ein Pionier der Computerspiele-Industrie mit über 30 Jahren Erfahrung und verantwortlich für zahlreiche Spiele-Klassiker hat BodyPaint 3D zum Bestandteil der Produktions-Pipeline gemacht.

Das japanische Studio mit Niederlassungen in aller Welt hat MAXONs 3D-Painting-Lösung intensiv bei der Produktion von 'Monster Hunter 3' eingesetzt, dem neuesten Zugang einer weiteren populären Gameserie.

BodyPaint 3D's UV-Werkzeuge und die hochentwickelten Malfunktionen wurden für die Texturierung zahlloser Monster, Charaktere und Waffen verwendet. Die Möglichkeiten, direkt auf der 3D-Oberfläche zu malen und mit Multi-Layer-Selektionen arbeiten zu können halfen dabei, den gewünschten Look zu erreichen.

Worauf also warten? Laden Sie Ihre kostenlose Demoversion noch heute!

BODYPAINT 3D
RELEASE 4.5

MAXON | 3D FOR THE REAL WORLD

WWW.BODYPAINT3D.DE

Wir können auch online!
www.makinggames.de

Zugegeben, es hat lange gedauert. Umso glücklicher sind wir jetzt mit dem Ergebnis: Die neue makinggames.de ist fertig und soll künftig eine Ihrer wichtigsten Anlaufstellen im Netz werden. Wir haben uns bemüht, die modernen Vorteile des Internets mit den traditionellen Stärken des Making Games Magazins zu verbinden. So haben wir unser Firmenregister der deutschen Spielebranche nicht einfach nur online gestellt, sondern es darüber hinaus mit einer Suchfunktion und Google-Maps-Einbindung erweitert sowie sämtliche Einträge unseres Branchenverzeichnisses Key Players als PDF integriert. Die Praktikumsbörse und unser Eventkalender bekamen ebenfalls ein ordentliches Internet-Facelifting verpasst. Und im Magazinbereich finden Sie inzwischen Dutzende unserer bislang veröffentlichten Fachtexte als Online-Version. Was uns dabei ganz wichtig ist: Heft und Internetseite sollen nicht miteinander konkurrieren, sondern sich gegenseitig ergänzen. Was das bedeutet, sehen sie rechts und in vielen Artikeln dieser Ausgabe. Überall dort, wo unsere Online-Plattform ergänzende, vertiefende oder auch widersprechende Informationen und Fachtexte anbietet, haben wir dies am Artikelanfang entsprechend gekennzeichnet. Natürlich haben wir noch viele weitere Ideen für makinggames.de, die wir in den nächsten Monaten umsetzen wollen. Und wie immer sind wir dankbar für jede Anregung unserer Leser: Was haben Sie sich schon immer von einer Branchen-Website gewünscht? Wie können wir uns weiter verbessern? Schreiben Sie es uns an info@makinggames.de.

Kleine Teams, große Erfolge

Das Online-Thema begleitet uns auch in unserer Titelstory »Digitale Distribution«. Gerade für deutsche Entwickler könnte dieser Vertriebsweg in den nächsten Monaten und Jahren enorm an Bedeutung gewinnen. Denn die Erfolge bei Browser-, Handy- und DS-Spielen machen eins deutlich: Deutsche Entwickler sind international immer dann vorn dabei, wenn es weniger auf Team- und Budgetgröße ankommt, sondern auf Können und Kreativität. Mit Ion Assault ist jüngst das erste deutsche Spiel für Xbox Live Arcade erschienen. Ein erster Beweis, dass diese zugegebenermaßen etwas lokalpatriotische These auch für Konsolenprojekte gilt. Und eine ebenso umfangreiche wie spannende Case Study, die Ihnen ab Seite 8 hoffentlich viele wertvolle Informationen über die Chancen und Risiken bei der Entwicklung von Download-Spielen liefert.

Viel Spaß beim Lesen wünscht

das Team von Making Games

Heiko Klinge

ist Projektleiter vom Making Games Magazin.

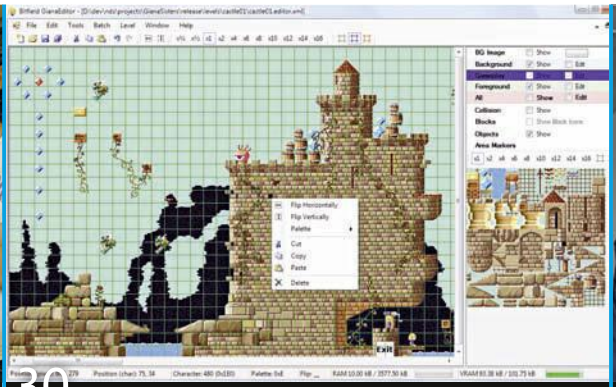
WWW
makinggames.de

- Immer wenn Sie diesen Hinweis bei einem Artikel sehen, möchten wir Sie auf ergänzende oder vertiefende Informationsangebote unseres Online-Portals aufmerksam machen.

»Heft und Internetseite sollen nicht miteinander konkurrieren, sondern sich gegenseitig ergänzen.«

04/2009
Inhalt

72 Brasilien – der schlafende Games-Riese?


30 Case Study: Effiziente Entwicklung auf dem Nintendo DS.

46 Martin de Rondes Charity-Publisher.

54 Noch einmal mit Gefühl: Ein wissenschaftlicher Essay über die emotionalen Probleme von Spielen.

16 Zehn Regeln für die Entwicklung von Download-Spielen.

Branche

06 **Branche / Köpfe**
Bemerkenswertes aus der Branche

07 **Eventkalender**
Wichtige Branchenveranstaltungen

Titelstory Digitale Distribution

08 **Case Study** Abenteuer Xbox Live Arcade
von Dominik Mieth und Roman Pfrengle

16 **Best Practice** Download-Spiele für Konsolen
von Pete Walentin

20 **Interview** Im Gespräch
mit Dan Connors (Telltale Games)
von Heiko Klinge

Rubriken

03 **Editorial**

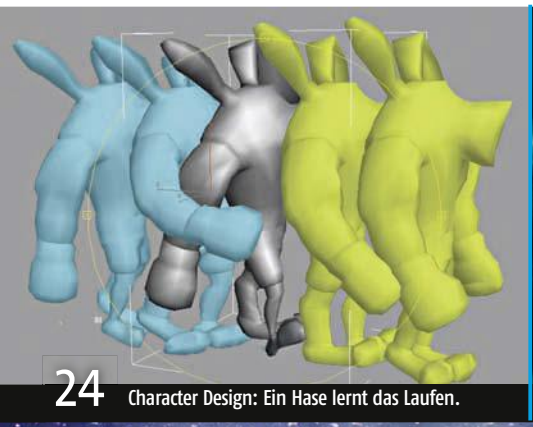
80 **Ein Tag bei ...** Radon Labs

82 **Vorschau**

82 **Impressum**

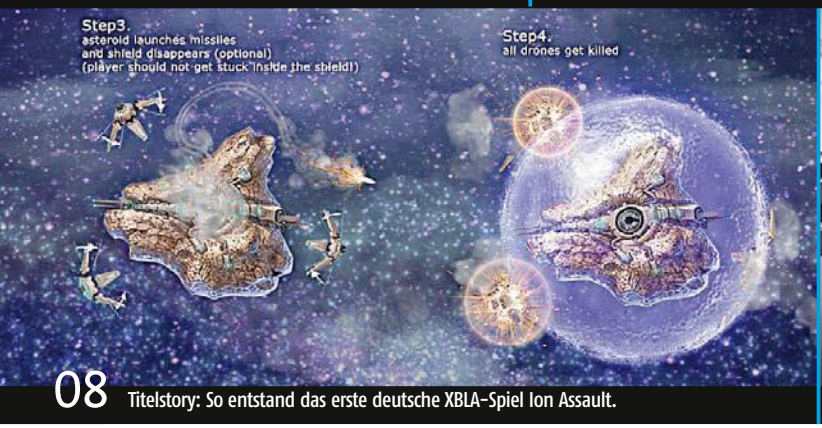
Grafik

24 **Workshop (Serie)** Character Design
von Chris Noeth


24 Character Design: Ein Hase lernt das Laufen.

36 Warum die Addon-Auslieferung in Eve reibungslos funktionierte.

60 Bewegungssteuerung auf dem PC.



08 Titelstory: So entstand das erste deutsche XBLA-Spiel Ion Assault.

64 Das Projekt MusicRacer.

40 Shader-Programmierung mit XNA.

Entwicklung

30 **Case Study**
Effiziente DS-Entwicklung
von Daniel Matzke und David Salz

36 **Case Study**
Addon-Auslieferung in Eve Online
von Kurt van Meter und James Wyld

40 **Programmierung**
Shader-Einbindung in XNA von Marc Ueberall

Business

46 **Publishing** Gründung eines Charity-Publishers
von Martin de Ronde

51 **Konferenz** Browsergames Forum
von Heiko Klinge

52 **Recht (Serie)** Entwicklungsverträge
von Claas Oehler

Campus

54 **Essay** Noch einmal mit Gefühl
von Björn Blankenheim

60 **Forschung** Geheimnisse der Bewegungssteuerung
von Philipp Wollmann

64 **Studentenprojekt** Musikspiel mal anders
von Stephan Krohn

68 **Ausbildung** Praktikumsbörse

Märkte

71 **Marktforschung**
GameStar- und GamePro-Leserdaten
von Yassin Chakhchoukh

72 **Überblick** Brasilien – eine Marktbetrachtung
von Andreas Stock

76 **Firmenregister** Die Branche im Überblick

GDC Europe 2009

Das nennt man wohl eine erfolgreiche Premiere: Mit rund 1.500 Besuchern schlug die erste GDC Europe die letzte Games Convention Developers Conference (circa 1.250 Teilnehmer) deutlich. Aber darf man es auch eine gelungene Premiere nennen? Da gingen die Meinungen der Besucher im Kölner Congress Center zumindest auseinander. Einige vermissten die familiäre Atmosphäre Leipzigs, andere kritisierten das Vortragsniveau. Beides ist durchaus nachvollziehbar: Im weitläufigen Areal traf man sich häufiger zufällig als geplant, gemütliche Treffpunkte wie die berühmte Leipziger Raucher-Terrasse fehlten. Und vor allem die Keynote-Sprecher enttäuschten, nutzten sie ihre Vorträge doch mehr zur Eigenwerbung als zur Wissensvermittlung. Crytek-CEO Cevat Yerli demonstrierte die technischen Vorzüge der CryEngine 3, CCP-Chef Hilmar Veigar Petursson kündigte sogar mit Dust 514 ein komplett neues Spiel an.

Dennoch bleibt festzuhalten: Die GDC Europe hat sich auf Anhieb als eine der wichtigsten und größten Branchenveranstaltungen Europas etabliert. Das Publikum war deutlich internationaler als in Leipzig, der Anteil amerikanischer und asiatischer Entwickler erkennbar höher. Dass weniger Platz zum Plaudern existierte, lag unter anderem auch daran, dass es anders als in Leipzig einen großen Ausstellungsbereich gab, auf dem sich über 40 Firmen wie Intel, Dolby und Autodesk präsentierten – für einige Unternehmen wie Scaleform (Interface Middleware) und Natural Motion (Animation) war es sogar die Deutschland-Premiere. Und ja, vom Vortragsniveau der GDC in San Francisco war die GDC Europe noch weit entfernt. Dennoch war viel Spannendes dabei, etwa das Panel »Designing Women« mit Kellee Santiago (Flower) oder die FAQ-Runde »Ask the Decision Makers«, auf der acht Entscheidungsträger von Publishern Einblicke in ihre Submission-Prozesse gaben.

Egal ob Kritiker oder Befürworter, in einem dürften sich alle einig sein: Die GDC Europe ist ein großer Gewinn für den Games-Standort Deutschland. Und das ist erst mal das Entscheidende. Bis zum nächsten Jahr. **Heiko Klinge**

Bayern startet Förderprogramm für Spiele-Entwickler

Der Geschäftsführer des FilmFernsehFonds Bayern **Prof. Dr. Klaus Schaefer** (links) und Medienminister **Siegfried Schneider** unterzeichnen das bayerische Games-Förderprogramm.

Genau 470.000 Euro – so viel will der bayerische Staat ab sofort jährlich in die Förderung von Computerspiel-Projekten investieren. Die Fördermittel können dabei nicht nur für die Prototypen-Erstellung beantragt werden, sondern auch bereits für die Konzeptentwicklung oder sogar für die eigentliche Produktion. Voraussetzung ist jedoch, dass die Spiele hohen qualitativen Ansprüchen genügen, maximal eine Altersfreigabe »ab 16« anstreben und weder sittliche noch religiöse Gefühle verletzen. Ego-Shooter dürften also eher schlechte Chancen haben. Über die Förderwürdigkeit entscheidet ein Vergabeausschuss des FilmFernseh-Fonds Bayern. Zum Gremium, das erstmals am 11. November tagen wird, werden neben Politikern und Pädagogen auch Experten aus der Spielebranche gehören, darunter Dr. Klemens Kundratitz (Geschäftsführer Koch Media) und Hendrik Lesser (Geschäftsführer Remote Control Productions). Förderanträge finden Sie auf: www.fff-bayern.de

Creativity World Forum 2009

Erstmals treffen sich 13 kreative Regionen aus aller Welt zu ihrem jährlichen Hauptevent in Deutschland und laden dazu auch ganz explizit die Spielebranche ein. Die Mischung aus Fachkongress, Ausstellung und Kooperationsbörse findet vom 30. November bis zum 3. Dezember in Stuttgart-Ludwigsburg statt. Mit Regisseurin Doris Dörrie, Nokia-Vorstand Esko Aho und MP3-Erfinder Prof. Dr. Dr. Karlheinz Brandenburg sprechen zwar nicht unbedingt ausgewiesene Games-Experten, aber ganz bestimmt interessante Personen. Und ein Vortrag wie »Kreatives Schreiben und Geschichten erzählen« (Doris Dörrie) könnte sicherlich auch für viele Game Designer spannend sein. Tickets für den Kongress kosten 60 bis 250 Euro. Wir sind als offizieller Medienpartner in jedem Fall vor Ort und verlosen darüber hinaus zwei Freikarten unter unseren Lesern. Wenn Sie beim Forum dabei sein möchten, schicken Sie einfach eine Mail mit Ihren Kontaktdaten an info@makinggames.de, Betreff: CWF 2009. www.cwf2009.de

Vier Fragen an ...

Elisabeth Mendoza ist Projektleiterin Digital Media bei der MFG Baden-Württemberg, Innovationsagentur des Landes für IT und Medien.

Making Games Sowohl im Ausstellungsbereich als auch unter den Sprechern finden sich kaum Vertreter aus der Spielebranche. Warum lohnt sich trotzdem der Besuch für die Leser des Making Games Magazins?

Elisabeth Mendoza Beim Creativity World Forum können sich Software- und Spielebranche hervorragend über die Möglichkeiten öffentlicher Förderung informieren. Darüber hinaus bringt das Event Unternehmen aus dem Bereich Games mit Vertretern aus Film, Simulation und Verlagswesen zusammen. So wird das Creativity World Forum zum Impulsgeber für neue, branchenübergreifende Kooperationen und Geschäftsmodelle. Gerade das Panel »Engineering & Digital Media« ist hier besonders interessant.

Making Games Vieles beim Forum dreht sich um das Thema Standortförderung. Wo sehen Sie da mögliche Anknüpfungspunkte für Unternehmen aus der deutschen Spielebranche?

Elisabeth Mendoza Da gibt es auf jeden Fall sehr viele Anknüpfungspunkte! Zum einen können Spielehersteller die Veranstaltung für Aufklärungsarbeit nutzen. Sie können den hiesigen Standort international bekannter machen und verschiedenen Entscheidern und Multiplikatoren die Potenziale digitaler Spiele aufzeigen. Zudem können Experten den spezifischen Förderungsbedarf in den Regionen diskutieren und passende Strategien entwickeln.

Making Games Von einigen Ausnahmen wie Gameforge einmal abgesehen, ist die Spielebranche in Baden-Württemberg derzeit eher unterrepräsentiert – gerade im Vergleich zu Standorten wie Hamburg, Berlin und der Rhein-Main-Region. Mit welchen Maßnahmen wollen Sie dies ändern?

Elisabeth Mendoza Da muss ich Ihnen widersprechen: In Baden-Württemberg gibt es mehr als 50 erfolgreiche Games-Unternehmen. Die baden-württembergische Spielebranche ist sowohl im Markt als auch auf dem Creativity World Forum keinesfalls unterrepräsentiert, und darüber freuen wir uns.

Making Games Das Forum ist das Hauptevent des globalen »Districts of Creativity«-Netzwerks. Inwieweit können Unternehmen der Spielebranche davon profitieren?

Elisabeth Mendoza Indem sie beim Creativity World Forum Flagge zeigen und diese internationale Großveranstaltung nicht nur als Ideengenerator nutzen, sondern auch als Plattform verstehen, auf der sie ihre Anliegen vertreten können. Ob potenzielle Geschäftspartner oder politische Entscheidungsträger: Beim Creativity World Forum sind wichtige Ansprechpartner für die Spielebranche vor Ort. Unternehmen sollten die Chance zu Gesprächen und einer positiven Weichenstellung nutzen!

Eventkalender

WWW makinggames.de
- Online-Eventkalender mit Suchfunktion und Veranstaltungsinfos

OKTOBER 2009

Making Games Talents

Ort:	München, IDG-Gebäude
Termin:	24. Oktober
Thema:	Recruiting-Event
Kosten:	29 Euro
Homepage:	www.makinggames.de/talents

Scrum Master Workshop

Ort:	München, Büros von Remote Control Production
Termin:	28. bis 29. Oktober 2009
Thema:	Seminar
Kosten:	1.400 Euro
Homepage:	www.r-control.de

NOVEMBER 2009

Browsergames Forum 2009

Ort:	Frankfurt, CineStar Metropolis
Termin:	6. bis 7. November 2009
Thema:	Fachkonferenz
Kosten:	freier Eintritt für Browsergames-Entwickler, sonst 200 Euro
Homepage:	www.bgf2009.de

Gamecity Treff

Ort:	Hamburg, 13. Stock
Termin:	11. November 2009, 19 Uhr
Thema:	Regio-Summit, Branchen-Treff
Kosten:	Eintritt frei, Anmeldung erforderlich
Homepage:	www.gamecity-hamburg.de

Creativity World Forum 2009

Ort:	Stuttgart-Ludwigsburg, Forum am Schlosspark Ludwigsburg
Termin:	30. November bis 3. Dezember 2009
Thema:	Fachkongress
Kosten:	50 bis 300 Euro
Homepage:	www.cwf2009.de

DEZEMBER 2009

Deutscher Entwicklerpreis 2009

Ort:	Essen, Lichtburg
Termin:	9. Dezember 2009
Thema:	Preisverleihungsgala
Kosten:	39 bis 89 Euro
Homepage:	www.deutscher-entwicklerpreis.de

Game Connection Europe

Ort:	Lyon, Centre de Congrès – Cité Internationale
Termin:	8. bis 10. Dezember 2009
Thema:	Networking-Event
Kosten:	1.100 bis 4.500 Euro
Homepage:	www.game-connection.com

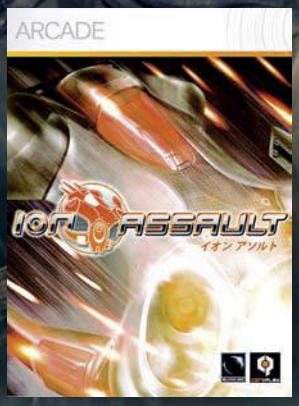
FEBRUAR 2010

Mobile World Congress

Ort:	Barcelona, Fira Montjuïc
Termin:	15. bis 18. Februar 2010
Thema:	weltweit größter Mobile-Fachkongress
Kosten:	599 bis 4.999 Euro
Homepage:	www.mobileworldcongress.com

Veranstalten Sie mit!

Sie organisieren eine Podiumsdiskussion oder planen eine Party und möchten, dass die Branche davon erfährt? Dann schreiben Sie uns an **info@makinggames.de**, **Stichwort »Eventkalender«** und wir veröffentlichen Ihre Veranstaltung in unserer nächsten Ausgabe.



Wichtig für den Ersteindruck: Das **Cover Artwork** von Ion Assault für den Xbox Marktplatz.

Next-Gen-Shooter Ion Assault
Abenteuer Xbox Live Arcade

Wie das erste deutsche Spiel für Xbox Live Arcade entwickelt wurde und mit welchen Besonderheiten man sich bei der Entwicklung eines Spiels für Microsofts Vorzeigeplattform auseinandersetzen muss.

Dominik Mieth

ist Game Designer bei Coreplay.

Dominik hat in Mainz Filmwissenschaft, Germanistik und Philosophie studiert und arbeitet seit 2006 als Game Designer und Producer in der Spielebranche. Seit Anfang 2008 bei Coreplay angestellt, entwickelt er primär Konzepte für Konsolenspiele auf Nintendo Wii, Xbox 360 und PlayStation 3. Derzeit arbeitet er an Coreplays drittem Projekt für Nintendo Wii.

Roman Pfrengle

ist Artist bei Coreplay.

Roman hat Kommunikationsdesign in Mainz studiert und arbeitet seit 2004 als 2D- und 3D-Artist. Unter anderem war an der Entwicklung von Spellforce: Shadow of the Phoenix und Spellforce 2 beteiligt. Er hat die Entwicklung von Coreplays FX-Editor von Anfang an begleitet und ist seitdem verantwortlich für sämtliche Spezialeffekte. Derzeit erstellt Roman Effekte und Grafiken für Coreplays drittes Projekt für Nintendo Wii.

WWW
makinggames.de

- Multiplattform-Entwicklung: Zehn Regeln für problemloses Portieren

Digitale Distribution hat einen wesentlichen Vorteil: Sie spart Kosten. Anders als bei Spielen, die im Ladenregal stehen, muss hier nicht vorher eine bestimmte Stückzahl mit Handbuch, Verpackung und Datenträgern produziert und vertrieben werden. Beim Download vervielfältigt sich das Spiel sozusagen selbstständig beim Verkauf, und es wird kein Kapital durch die physische Vervielfältigung gebunden. Auch der Vertrieb fällt weg, denn der Download ist automatisch überall dort verfügbar, wo ein Internetanschluss vorhanden ist. Dafür entstehen natürlich andere Probleme wie zum Beispiel die Abrechnung oder der Kopierschutz (Digital Rights Management).

Hier kommen etablierte Downloadportale wie Steam, Impulse, PlayStation Network oder Xbox Live Arcade (XBLA) ins Spiel. Sie bieten einen weiteren Vorteil: Sie bringen bereits ein Stammpublikum mit, das automatisch auf Neuerscheinungen aufmerksam gemacht wird: Wer sich einloggt oder die Konsole einschaltet, sieht automatisch, was es Neues gibt und kann in den meisten Fällen direkt eine Testversion laden oder Bewertungen anderer Spieler einsehen.

Warum Xbox Live Arcade?

Vor diesem Hintergrund ist schnell erklärt, warum Ion Assault, der erste XBLA-Titel aus Deutschland, ein Downloadspiel werden musste. Im März 2007 existiert Coreplay nur als Idee von Peter Ohlmann und Andreas Drude. Coreplay sollte als unabhängiger Entwickler neuartige Spielkonzepte visuell ansprechend umsetzen – und zwar nicht wie die meisten Entwickler in Deutschland nur für den PC, sondern auch für Konsolen und wenn irgend möglich über die Grenzen Europas hinaus.

Zu diesem Zeitpunkt hat Coreplay bereits den ersten Pitch hinter sich: einen ambitionierter Multiplayer-Shooter für PC und 360. Es existiert ein Technik-Prototyp, der eine eindrucksvolle Terrain Engine demonstriert, aber nur sehr rudimentäres Gameplay. Genau das aber wollen Publisher sehen, bevor sie mit einem Studio in Gründung zusammenarbeiten. Doch ein Spielprinzip umzusetzen, kostet Geld. Um also einen Prototyp zu präsentieren, den die Publisher gern sehen würden, wäre eine Finanzierung nötig, die aber wiederum einen Geldgeber erfordert. Ein bekanntes Henne-Ei-Problem.

Hier kommt die Idee auf, ein kleineres Projekt zu machen. Ein Projekt, das einfach zu prototypen ist und dennoch zeigt, was das Team und seine Technologie zu leisten vermögen. Es drängt sich Xbox Live Arcade als Plattform auf, einfach weil die Xbox 360 zu diesem Zeitpunkt die einzige Konsole mit hauseigener Downloadplattform ist, die man damals schon in Deutschland kaufen kann. Die PlayStation 3 wird erst einige Monate später in Europa erscheinen. Wii-Ware für Nintendos Wii liegt in noch fernerer Zukunft. Ein Spiel komplett eigenständig zu entwickeln und auf dem PC zu vertreiben, steht nicht zur Diskussion.

2006: Xbox Live Arcade als Chance

Jeff Tunnell (CEO von GarageGames) hat zu dieser Zeit auf seinem Blog »make it BIG in games« bereits einige interessante Eckdaten geschildert, die auf seiner Erfahrung mit Marble Blast Ultra basieren: 100.000 bis 300.000 US-Dollar sollen die aktuellen Arcade-Spiele in der Entwicklung kosten. Und selbst Titel, die eher schleppend laufen, erzielen nach diversen Schätzungen Verkäufe im fünfstelligen Bereich.

Diese Schätzungen basieren meist auf einer Kontrolle der Einträge in den Highscore-Listen, auf die nur die Spieler einer Vollversion Zugriff haben. Genaue Zahlen gibt es nicht, aber die Aussicht bleibt: Mit dem Geld, das man bräuchte, um den Prototyp für ein »großes« Spiel zu entwickeln, lässt sich hier eine ganze Produktion finanzieren!

Auch ein zweiter Aspekt ist interessant: Für Live Arcade kann man Spielkonzepte entwickeln, die sich außerhalb der ausgetretenen Genrepfade handelsüblicher PC-Spiele bewegen. Hier wird der Markt nicht in erster Linie durch die üblichen Verdächtigen bestimmt. Die Spieler freuen sich hingegen vielmehr über ein Wiedersehen mit Klassikern, sowie die Wiederbelebung von Genres und Spielkonzepten aus der 8- und 16-Bit-Ära und scheinen generell aufgeschlossener und experimentierfreudiger zu sein.

Vom ersten Konzept ...

Am Abend des 13.3.2007 sitze ich zum ersten Mal mit Andi und Peter in Mainz zusammen. Auch Roman Pfrengle ist dabei, der später für Ion Assault die Partikeleffekte erstellen wird. Peter zeigt uns einige Technologiedemos, die er mit seiner Engine erstellt hat, darunter auch eine Partikelphysiksimulation, die auf der damals aktuellen Geforce-6-Hardware läuft und deren Shader Pipelines zum Glühen bringt. Zu viert überlegen wir uns bis spät in die Nacht Spielkonzepte. Dabei wird die erste Idee erst verworfen, taucht dann aber doch wieder auf: Ein klassisches Asteroids-Spiel, in dem man nicht mit einer Kanone auf die Asteroiden im Dauerfeuer ballert, sondern die Asteroiden mit den beweglichen Partikeln erhitzen muss. Die Idee bleibt hängen; nicht zuletzt, weil sie sich so schön in einem Satz zusammenfassen lässt. Außerdem folgt sie in weiten Teilen einem erprobten Spielkonzept, das seit den 70er-Jahren unverwüstlich ist, stellt dieses Prinzip aber durch einen simplen Kniff total auf den Kopf: Partikel bündeln statt Dauerfeuer.

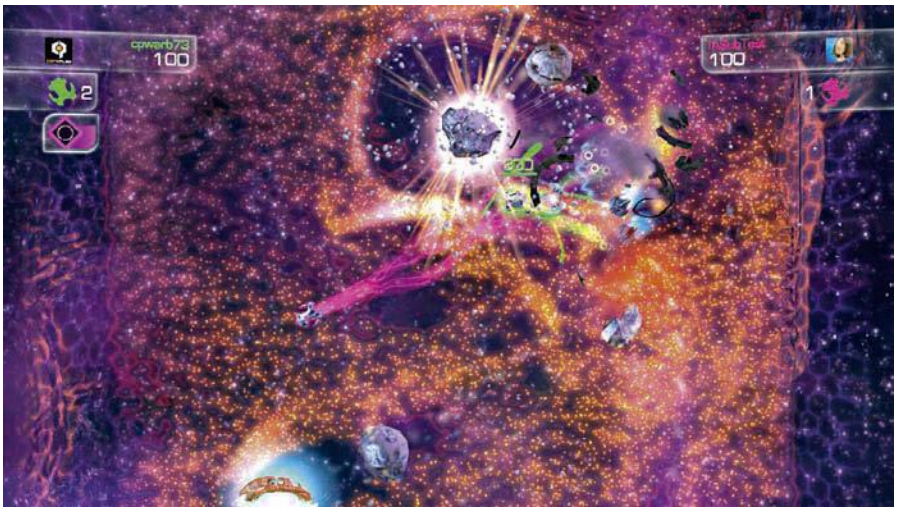
Das Schöne dabei: Der Kniff ist tatsächlich ein Spielelement, das erst durch die moderne Hardware ermöglicht wird. Wo sich 2007 noch die ersten Spiele der »Next Generation« gefallen lassen müssen, dass sie vom Gameplay her nicht mehr bieten als ihre direkten Vorgänger auf der vorangegangenen Hardware, können wir mit Fug und Recht behaupten, dass es sich um ein echtes »Next Gen«-Erlebnis handelt.

... über Prototyp und Pitching ...

Dank der bereits vorhandenen Technik gibt es schon wenige Tage später einen spielbaren Prototyp. Dieser demonstriert die wesentlichen Spielelemente: Man kann ein Raumschiff steuern, damit die Partikel aufsammeln und alle gesammelten Partikel mit einem Schuss auf Asteroiden abfeuern, die bereits effektvoll explodieren und in kleinere Bruchstücke zerfallen.

Das Spiel wird Ion Assault getauft, Pitching-Material wird zusammengestellt und auf der

August 2007: Der **finale Prototyp** enthält neben der Partikelphysik bereits erste Spezialeffekte und zwei Gegner.



Juli 2009: Im **fertigen Spiel** ist alles bis zum Anschlag aufgebohrt, optisch ergänzt ein Fluidsystem die fertigen Partikeleffekte. Die meisten visuellen Effekte erkennt man allerdings erst in Bewegung.

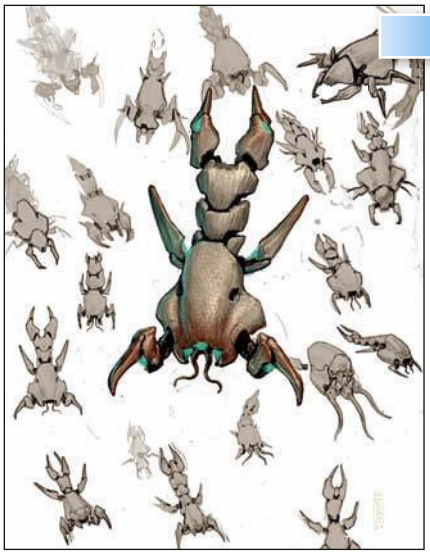
Games Convention im August 2007 gezeigt. Es gibt nur ein Problem: Der Prototyp läuft noch auf dem PC und benötigt einen einigermaßen potenten Rechner, der einer 360 ebenbürtig ist und mindestens eine Geforce 7 enthält – eine Leistungsklasse, die die meisten Laptops zu dieser Zeit nicht erreichen. Vor Ort können wir daher nur Screenshots und eine Powerpoint-Präsentation zeigen. Das wirklich Spektakuläre an der Grafik – Tausende Partikel in Bewegung – geht hier verloren. Außerdem lässt sich das Einzigartige am Spielprinzip nur sehr schwer erklären. Die meisten, die einen Screenshot sehen, erkennen darin nicht mehr als einen Klon von Geometry Wars. Erst Leute, die den Prototyp spielen, registrieren den Unterschied.

... zum Produktionsstart

Es dauert über ein Jahr, bis wir im Herbst 2008 mit RTL Interactive einen Publisher für Ion Assault gefunden haben. Inzwischen sind wir registrierter Entwickler für Xbox 360 und Ninten-

Coreplay

Coreplay wurde 2007 von Peter Ohlmann und Andreas Drude gegründet. Dank einer intern entwickelten Engine ist Coreplay als lizenzierter Entwickler für Microsoft Xbox 360, Sony PlayStation 3 und Nintendo Wii in der Lage, sehr effizient Spiele für Konsolen und PC parallel zu entwickeln. Einzelne Bestandteile der Coreplay Engine, wie die leistungsfähige Coreplay FX-Engine, wurden bereits von anderen Studios lizensiert. www.coreplay.de



Scribbels und Formstudien zu einem Gegner.

Rendering des Models mit hochauflösender Textur. Die Texturgröße wird für die Verwendung im Spiel reduziert.

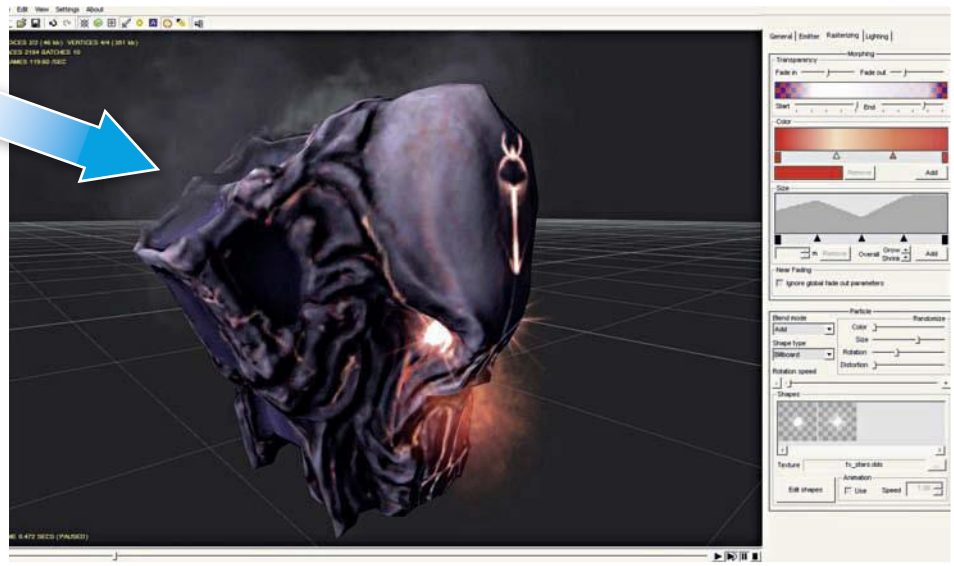
do Wii und haben auf beiden Plattformen erste Erfahrungen gesammelt. Die Coreplay Engine und die dazugehörigen Tools laufen auf beiden Plattformen, und die Tatsache, dass wir auch Ion Assault inzwischen auf der 360 zeigen können, hat mit Sicherheit wesentlich dazu beigetragen, Bedenken seitens des Publishers zu zerstreuen.

Die einzige Bedingung von RTL ist, dass wir das Konzept um einen online spielbaren Multiplayer-Teil erweitern. Das ist durchaus eine problematische Forderung, weil sich der Singleplayer-Part nicht so einfach erweitern lässt: Positionen, Geschwindigkeit und Bewegungen von Tausenden Partikeln sind entscheidend für das Spielprinzip und lassen sich unmöglich über die beschränkte Bandbreite der meisten Internetverbindungen synchronisieren. Also wird der Multiplayer-Modus ein zusätzliches Spiel, das wir neben der Einzelspielerkampagne entwickeln. Auch hier steht ein Klassiker Pate. In diesem Fall ist es Spacewar!, das wir um Spielerbasen erweitern, die man verteidigen muss, während man gleichzeitig versucht, mit den Schiffen, die in der eigenen Basis produziert werden, die Basis der Gegner anzugreifen. Erweitert um einen Mehrspielerpart geht Ion Assault schließlich im Oktober 2008 in die acht Monate lange Produktion.

2008 / 2009: Konkurrenz auf hohem Niveau

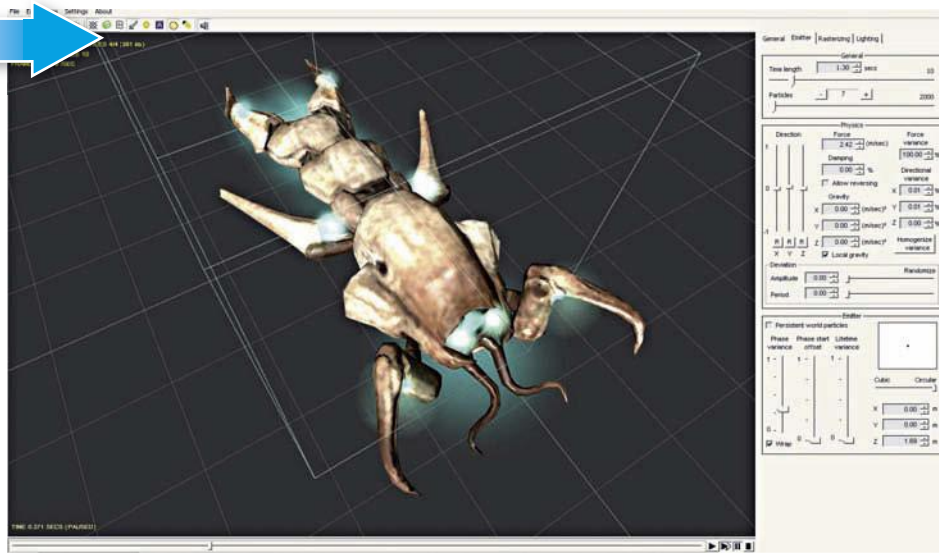
Die Konkurrenz auf Live Arcade ist inzwischen eine andere: Längst haben große Publisher das Potenzial von Arcade-Spielen erkannt. Wo 2006 noch UNO und neu aufgelegte Klassiker wie Street Fighter 2 die Verkaufscharts anführen und 2007 immer noch Remakes und Emulationen von Klassikern wie Turtles in Time und

Die **Konzeptzeichnung** für die »Asteroiden« im dritten Setting, in dem der Spieler Ruiner erforscht.



Ein Säulenteil als **fertiges Modell** mit diversen Effekten, die jeweils einzeln am Objekt platziert werden. Auch das Leuchten des Symbols auf der Stirn ist beispielsweise als Spezialeffekt umgesetzt.

Das fertige Modell mit reduzierter Textur und Beleuchtung im Viewer.



Das Spielmodell wird im **Effekteditor** anschließend mit Effekten versehen.

Worms dominieren, erobern 2008 die ersten primär für die Plattform entwickelten Spiele die Spitzenplätze: Castle Crashers, Geometry Wars 2 und Braid. Die Wertigkeit der Produktionen ist gestiegen, mit ihr die Budgets, und dass jede Woche zwei Spiele erscheinen, ist nun ebenfalls zur Regel geworden.

Braid-Entwickler Johnathan Blow hat seine Entwicklungskosten noch auf 200.000 US-Dollar geschätzt. Shadow Complex, das die Unreal Engine 3 nutzt, dürfte ein Vielfaches gekostet haben. Die Konzepte auf der Plattform sind vielleicht kompakter, die Umsetzung auf einem solchen Niveau ist aber nur noch von erfahrenen Entwicklern zu leisten. Spiele, die dem optischen Vergleich mit solchen Spitzentiteln nicht standhalten, laufen Gefahr, bereits im Ersteindruck beim Spieler durchzufallen.

Durch die gestiegene Konkurrenz und das große Angebot sind laut den Schätzungen auf VGChartz.com die Verkaufszahlen einzelner Titel inzwischen an der 10.000-Stück-Grenze gescheitert. Kurz: Die Pionierzeit, in der eine Emulation des Automaten Frogger sechsstellige Verkaufszahlen erzielt, ist vorüber. Heute sehen die Bedingungen auf der Plattform anders aus:

- Spiele wie Shadow Complex haben die Messlatte für den Produktionswert angehoben. Von einem XBLA-Spiel wird heute erwartet, dass es qualitativ einem »boxed« Vollpreisspiel entspricht. Die maximale Downloadgröße ist inzwischen von einst 50 Megabyte auf 2 Gigabyte angestiegen.
- Indie-Spiele auf XBLA sind die Ausnahme. Wer auf die Plattform will, muss sich mit den Spielen von erfahrenen Teams wie DICE (Battlefield 1943) oder Valve (Portal: Still Alive) vergleichen und mit emulierten Klassikern wie Sonic The Hedgehog konkurrieren.
- Der regelmäßige Nachschub an Titeln verkürzt enorm die Zeit, in der ein Spiel seine Produktionskosten einspielen kann. Nur wenige Titel genießen weiterhin die Aufmerksamkeit der Spieler, nachdem sie aus den vorderen Plätzen der »Neu erschienen«-Übersicht verschwunden sind.
- Ähnlich wie bei Vollpreisspielen handelt es sich bei vielen der erfolgreichen Spiele um Fortsetzungen mit bekannten Namen.

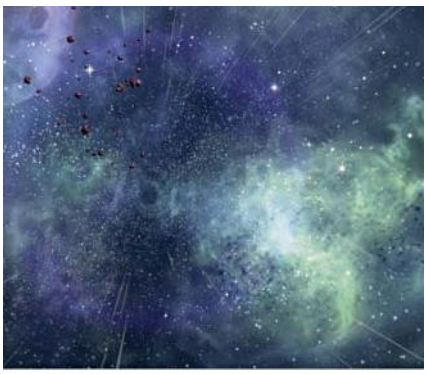
Insgesamt gibt es kaum noch Vorteile gegenüber der Entwicklung eines Vollpreis Konsolenspiels. Einer Entwicklung, der Microsoft inzwischen durch die Einführung der Indie-Games Rechnung getragen hat, die jedoch für professionelle Entwickler keine echte Alternative darstellen.

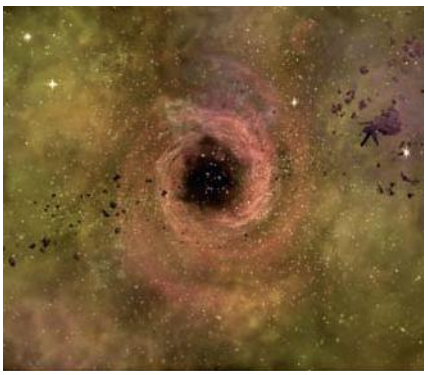
Sich technisch und inhaltlich von der breiten Masse abheben

Letzten Endes stehen Entwickler, die ein Spiel für die Plattform Xbox Live Arcade entwickeln wollen, also vor dem gleichen Problem wie Entwickler, die ihr Spiel in einer Schachtel im Einzelhandel sehen wollen: Sie müssen einen Publisher als Partner finden, der von den Qualitäten ihres Produkts überzeugt ist und es finanziert. Anschließend braucht er zwar keine Datenträger pressen und Schachteln stanzen lassen, muss aber dafür einen Slot auf Microsofts angesagter Plattform ergattern.

Hier konnten wir bei Ion Assault bereits im Prototyp-Stadium zwei Stärken ausspielen, die dazu beigetragen haben, RTL Interactive Games Publishing von dem Spiel zu überzeugen: Ion Assault hebt sich von der Spielmechanik klar von allen bisher auf der Plattform erschienenen Spielen ab, und Ion Assault ist in der Lage, die meisten Konkurrenten grafisch in ihre Schranken zu weisen.

Die Bedeutung des grafischen Ersteindrucks kann dabei gar nicht genug betont werden: Als im Juli 2009 erste Trailer und Bilder von Ion Assault im Internet veröffentlicht werden, ist die





Beispiele für **Fullscreen-FX-Collagen**, wie sie in Ion Assault bei den Levelübergängen und während des Resultscreens verwendet werden.

Konzept und Ablauf eines **Bosskampfs**: Die Zerstörung des Asteroid Fort in mehreren Stufen.

Grafik der wesentliche Faktor, der – gerade international – Aufmerksamkeit für das Spiel generiert. Im Folgenden wird unser FX-Artist Roman Pfrengle die Erstellung der Partikeleffekte für Ion Assault beschreiben, die einen wesentlichen Bestandteil des optischen Eindrucks ausmachen.

Eigenschaften der FX-Engine

Da Ion Assault ein sehr effektlastiger Titel ist, war es wichtig, von Anfang an auf ein komfortables und effizientes FX-Tool wie die Coreplay FX-Engine zurückgreifen zu können. Die erste Version des Tools entstand bereits im Jahr 2005 unter dem Namen ShaderMagic. Grundgedanke war schon damals, ein Tool zu erschaffen, das ebenso benutzerfreundlich wie effektiv ist. Es sollte mit einer überschaubaren Anzahl von Parametern möglichst viele verschiedene Effektvariationen abdecken. Dadurch spart einerseits der Programmierer viel Zeit, weil er keine Effekte mehr gezielt ausprogrammieren muss. Andererseits hat der Grafiker die volle visuelle Kontrolle über die Effekte. Kenntnisse von umständlichen Script- und Programmiersprachen sind dabei nicht nötig. Das Tool konnte zudem in kurzer Zeit speziell für die Anforderungen von Ion Assault angepasst und erweitert werden.

Die Idee, extrem viele imposante Effekte einzubauen, existierte von Anfang an. Effekte sind plakativ und gut sichtbar, während – vor allem bei einem Vogelperspektiven-Spiel wie Ion Assault – bei Texturen und Materialien Details schnell verloren gehen. Daher besitzt jedes Objekt ein komplettes Set an individuellen Effekten. Absoluter Standard für jeden Gegner ist ein Spawn-FX (Gegner kommt aufs Spielfeld), ein geloopter Idle-FX (z.B. Triebwerke, leuchtende Elemente passend zur Textur etc.) sowie ein Explosions-FX. Zusätzlich können Gegner aber eine Vielzahl weiterer Effekte besitzen, wie etwa ein Schussprojektil, Schussvorankündigung, Auftreffen des Schusses usw.

Durch spezielle Gegner wie die mehrgliedrigen Würmer war es außerdem nötig, Effekte an Joints platzieren und abspielen zu können. 3D-Objekte und deren Animationen mussten sich daher in die FX-Engine importieren lassen. Diese Funktion ermöglichte uns das passgenaue Erzeugen von Effekten für jedes vorhandene statische oder animierte Mesh.

Große Effekte in mehreren Stufen mit vielen Einzelelementen

Erst durch das Überlagern mehrerer Bestandteile in mehreren Schichten und Stufen mit abgestimmten Timings wird ein Effekt wirklich optisch ansprechend. Das gilt ebenso für Explosionen, die als echtes Effektfeuerwerk mit Anfang und Ende orchestriert werden, wie für die Trail-FX: Eine Einheit kann durchaus einen Trail-FX haben, der aus fünf Effekten besteht, die an verschiedenen Joints gelagert sind. Die Einheit »Echelon Juggernaut« hat zum Beispiel vom Erscheinen bis zur kompletten Zerstörung mit verglühenden Bruchstücken insgesamt 15 Effekte, die wiederum aus einer Vielzahl einzelner Elemente bestehen. Ein Endgegner wie das »Asteroid Fort« besitzt sogar mehr als doppelt so viele.

Die einzelnen Shapes, aus denen ein einzelnes FX-Element entsteht, werden dabei aus thematisch sortierten Texturkatalogen geladen. Dadurch können einem Element mehrere Shapes zugewiesen werden. Verschiedene Regler für Größe, Farbe, Drehung, Kräfte, Geschwindigkeiten und deren Zufallswahrscheinlichkeit lassen FX Gestalt annehmen. Für manche organischen FX werden dabei aus Maya Fluids-Animationen in Einzelframes herausgerechnet, die dann in der FX-Engine animiert werden. FX werden aber nie komplett vorgerendert, da dies in der Masse einen zu statischen Eindruck hinterlässt. Dies wird durch ein Zufallselement verhindert, das einzelne kleine Bestandteile wie etwa Funken in Position und Anzahl variieren kann.

Eine weitere Besonderheit der Coreplay FX-Engine ist, dass sie sehr große und aufwändige Partikeleffekte zulässt. So sind das gesamte Menü (ausgenommen die UI) warp-in/warp-out sowie die Endsequenzen der Levels ein riesiger Effekt, der wiederum aus vielen Einzelelementen besteht. Alles in allem hat Ion Assault dadurch über 750 größtenteils sehr einzigartige und aufwändige Effekte. Dass all diese Effekte in einer Zeitspanne von lediglich drei Monaten von nur einer Person erstellt werden konnten, demonstriert die Leistungsfähigkeit des Tools.

Balanceakt zwischen Effektfeuerwerk und Übersicht

Die spezielle Herausforderung im Einsatz von Effekten bei Ion Assault liegt darin, dass durch die Spielmechanik bereits zahlreiche Partikel auf dem Spielfeld sind, die sich zudem meist bewegen und dabei ihre Farbe ändern. Die restlichen Partikeleffekte von Objekten, Gegnern und Power-Ups stehen dadurch immer in Konkurrenz mit den Gameplay-relevanten Partikeln.

Viele FX wirken im Spiel daher anders als im Editor der FX-Engine. Die Feineinstellung des FX geschieht somit nicht unbedingt im Editor, sondern muss nach genauer Begutachtung im Spiel erfolgen. Ein Effekt muss aufdringlich genug sein, um wahrgenommen zu werden, darf aber andererseits auch nicht zu plakativ ausfallen, um die Übersichtlichkeit nicht zu sehr zu beeinträchtigen. Ob es sich um reines »Eyecandy« oder einen funktional wichtigen FX handelt, spielt eine große Rolle. Wichtige FX müssen sich stärker gegen die Spielpartikel behaupten als reine Schmuckeffekte. Eine wesentliche Maßnahme war hier, dass jedes Setting einem festen Farbschema folgt, das fast ausschließlich komplementär ausfällt. Konzeptgrafiken, die das Aussehen, den Antrieb, die Schüsse und die Explosionen beschreiben und dabei bereits dem Farbschema Rechnung tragen, waren hier extrem hilfreich.

Wer spielt Trial und wer kauft die Vollversion?

Während Effekte und Partikelphysik als wortwörtlicher Blickfang Aufmerksamkeit generieren können, müssen wir, was das Gameplay betrifft, auf eine weitere Eigenschaft von Xbox Live Arcade vertrauen: die obligatorische Trial Version. Wir wissen, dass jeder das Spiel kostenlos ausprobieren kann und damit eine Gelegenheit bekommt, auch die Spielmechanik zu verstehen. Im Hinblick auf die Trial gibt es allerdings eine Besonderheit zu beachten: Hier handelt es sich nicht um eine beschränkte Demoversion, sondern in der Tat um die Vollversion, die der Spieler herunterlädt und in der letzten Endes nur ein Bit geändert werden muss, um aus der Trial das volle Spiel zu machen.

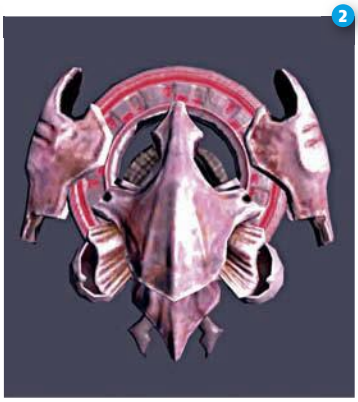
Für die Spieler heißt das, dass sie niemals Gefahr laufen, die Katze im Sack zu kaufen. Für die Entwickler heißt das aber auch: Das Spiel muss von Anfang an gut sein und den Spieler fesseln.

Eine Woche im Rampenlicht

Hier zeigt sich eine mögliche Schwäche von Ion Assault: Ein dermaßen eigenständiges Spielprinzip erfordert Einarbeitungszeit. Wir haben deshalb in der zweiten Hälfte der Entwicklung viel Wert darauf gelegt, das Spiel immer wieder von Testern spielen zu lassen, die ins kalte Wasser geworfen selbstständig mit der Steuerung zurechtkommen müssen. Hier hat sich schnell eine Tendenz offenbart:

Spieler gehen mit einer festen Erwartungshaltung an ein Spiel heran. Sie ordnen es etwa einem Genre zu und stellen es damit in direkten Vergleich mit anderen Spielen aus diesem Bereich. Im Fall von Ion Assault waren das die sogenannten Twin-Stick-Shooter, die durch das Genre-Flaggschiff Geometry Wars bekannt gemacht wurden. Hier mag im direkten Vergleich das oberflächliche Spielprinzip (Gegner und Objekte abschießen) übereinstimmen, die Spielmechanik ist aber eine grundsätzlich andere. Partikel sammeln und alle gesammelten Partikel in einem Schuss abfeuern steht dem Ausrichten einer Dauerfeuer-Kanone mit unendlichem Munitionsvorrat gegenüber. Das Ergebnis: In den ersten Minuten ist der Spieler irritiert, er muss umdenken, das Spielprinzip verstehen und die Steuerung verinnerlichen.

Eine Konzeptstudie des **Tau Ceti Defenders** – einer der schlafenden Wächter, deren Ruhe der Spieler bei seiner Ankunft stört.

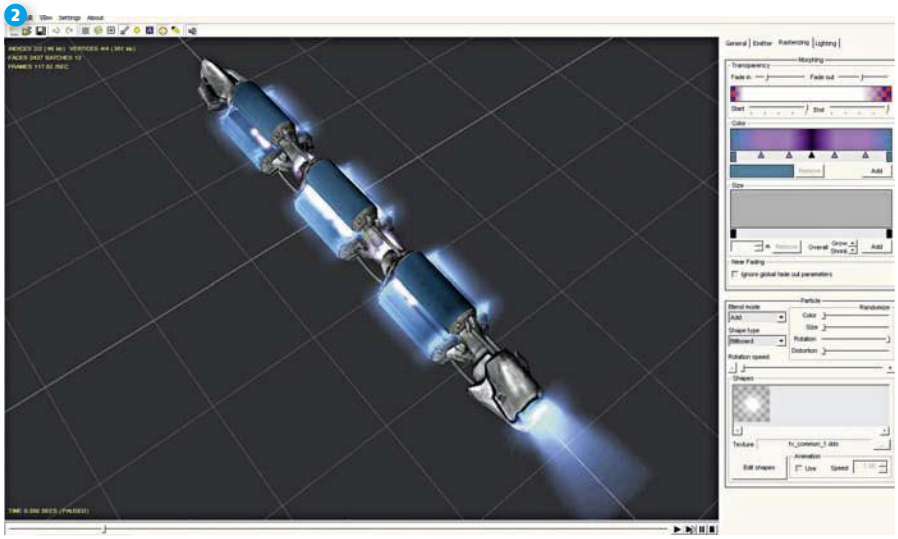


Der Tau Ceti Defender als fertiges **Mesh** mit Textur und Beleuchtung im Viewer …

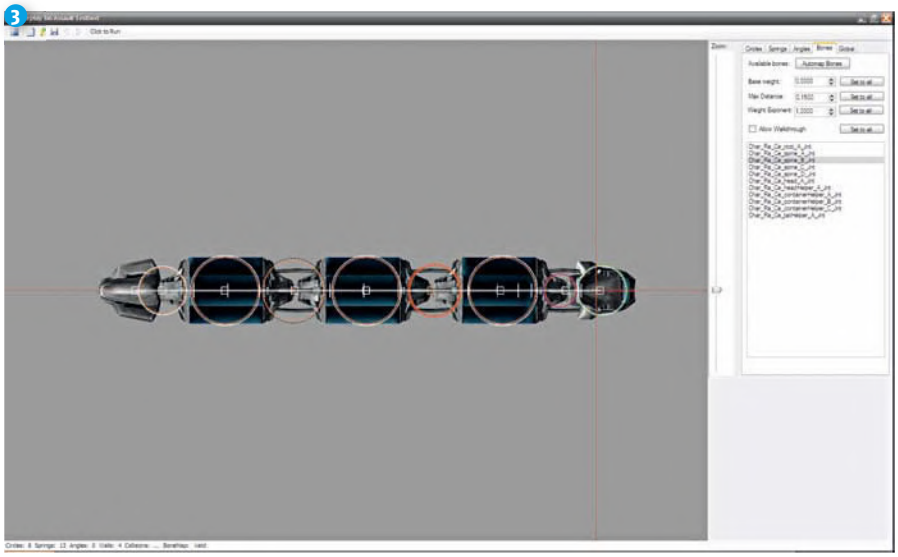
… und ausgestattet mit seinem loopbaren Idle Effect im FX-Editor.

Konzeptzeichnung für den **Echelon Juggernaut**, einen mehrgliedrigen Gegner …

Wir haben in den Testsitzungen genau auf die Uhr gesehen und festgestellt, dass es im Schnitt drei Minuten dauert, bis man Ion Assault grundlegend beherrscht. Knackpunkt ist hier die Erkenntnis, dass man die Partikel nicht nur aufsaugen und wieder wegschießen, sondern auch gesammelt vor sich hertragen und als Schild oder Nahkampfwaffe nutzen kann. Wir haben unser Möglichstes getan, die Zeit bis zu dieser Erkenntnis durch ein Tutorial und zusätzliche Anzeigen zu überbrücken. Maßnahmen waren beispielsweise ein Laser als Zielhilfe sowie ein Barometer, das anzeigt, wie stark ein momentaner Schuss mit den bislang gesammelten Partikeln ausfallen würde. Die Bereitschaft seitens des Spielers, sich in das Spiel einzufinden, bleibt aber Voraussetzung – insbesondere weil die ungewohnte Steuerung eng mit dem eigenständigen Spielkonzept verknüpft ist, auf dem alle Elemente basieren. Hier Hand anzulegen hätte bedeutet, nach und nach das gesamte Spiel umstricken zu müssen.



… der mit **Effekten** ausgestaltet …



… und im **Physik-Editor** mit physikalischen Eigenschaften versehen wird. Im Spiel reagiert der Echelon Juggernaut entsprechend realistisch auf Kollisionen, zum Beispiel mit Asteroiden.

Ob wir dem durchschnittlichen Spieler hier zu viel abverlangen, bleibt abzuwarten. Man kann aber davon ausgehen, dass in der aktuellen Situation auf Xbox Live Arcade solche Experimente riskant sind: Ion Assault muss sich bereits einen »Arcade Wednesday« und die Aufmerksamkeit mit zwei anderen Spielen teilen. Sollten die Konkurrenzprodukte zu viel Aufmerksamkeit auf sich ziehen, besteht die Gefahr, dass Ion Assault schon in der nächsten Woche wieder in Richtung Vergessenheit rutscht. Denn dann erscheinen die nächsten Arcade-Spiele und stehen im Rampenlicht. Bisher haben es nur wenige Spiele geschafft, sich zu Selbstläufern zu entwickeln. Nur Spielen, denen der Sprung in die Top-Verkaufslisten gelingt, bleiben im Dashboard über Wochen präsent, werden weiterhin heruntergeladen.

Von Submissions und Approvals

Jedes Konsolenspiel, unabhängig von der Plattform, muss eine technische Prüfung (Technical Certification) beim Konsolenhersteller erfolgreich absolvieren, um letztendlich überhaupt erscheinen zu dürfen. Dies garantiert dem Hersteller der Konsole eine ausreichende technische Qualität der Produkte. Bei Xbox Live Arcade sind einige Bezeichnungen wie zum Beispiel Menüpunkte vorgegeben, um dem Spieler ein vertrautes Umfeld anzubieten. Außerdem sollte natürlich jedes XBLA-Spiel die Vorteile der Plattform nutzen. Da wären unter anderem Achievements (Erfolge), Rich Presence (»Player XY spielt gerade …«), Leaderboards und das Onlinespiel via Xbox Live. Insbesondere die technisch einwandfreie Anwendung des Onlinespieles via Xbox Live stellt für auf dieser Plattform unerfahrene Programmierer eine Herausforderung dar. Hier ist das Aufsetzen eines sauberen Frameworks unabdingbar, falls man diese Funktionalität in zukünftigen Titeln erneut einsetzen möchte. Zum Glück ist der technische Support seitens Microsoft hier sehr gut und die notwendigen SDKs (Software Development Kits) sind ausführlich dokumentiert. Sobald das Spiel dann seitens des Entwicklers fertig gestellt wurde, findet die sogenannte Submission statt. Hier prüft Microsoft das Spiel auf Herz und Nieren und informiert den Entwickler über eventuelle Fehler oder Probleme. Der Entwickler sollte sich bereits im Vorfeld mit den TCRs (Technical Certification Requirements) auseinandersetzen, denn diese beinhalten diverse Testszenarien und alle notwendigen Funktionen im Spiel. Damit wird zum Beispiel garantiert, dass bei Problemen die korrekten Fehlermeldungen angezeigt werden oder aber auch, dass Spiele nicht zu lange im Ladescreen verharren und somit den Spieler verunsichern. Sobald alle TCRs erfüllt sind, das Spiel stabil und performant läuft und diverse Dokumente (z.B. Altersfreigaben) und Assets (z.B. Logos) korrekt eingereicht wurden, wird das Approval erteilt. Dieses Approval ermöglicht nun die Veröffentlichung des Spiels auf dem Marktplatz und beendet somit den Entwicklungsprozess.

Zusammenfassung: Vor- und Nachteile

Abschließend kann man die Vor- und Nachteile der Plattform XBLA wie folgt zusammenfassen:

Vorteile

- Auf Arcade erscheinen Spiele, die einen geringeren Umfang als Retail-Produkte haben, daher im Produktionsaufwand attraktiver sein können und in der Regel leichter zu prototypen sind.
- Die Hardware der Xbox 360 ist leistungsstark und erfordert unter Umständen weniger Kompromisse als beispielsweise im Download-Segment am PC, wo erfolgreiche Downloadspiele häufig für ältere Hardware optimiert werden müssen.
- Auf Arcade werden andere Genres akzeptiert als im Retail-Segment. Dazu gehören beispielsweise klassische 2D-Genres, die in 3D Grafik wiederbelebt werden.
- Auf Arcade werden teilweise noch Spielkonzepte mit einem geringeren Umfang akzeptiert. Viele Käufer haben nichts dagegen, ein Spiel vor dem Erscheinen der nächsten Arcade-Spiele abgeschlossen zu haben.
- Der »Arcade Wednesday« ist etabliert, und jedes Spiel, das auf Arcade erscheint, erhält die gleiche Aufmerksamkeit im Dashboard.
- Arcade-Spiele erscheinen in den jeweiligen Territorien weltweit gleichzeitig und sind so einer breiten Käuferschicht zugänglich.

Nachteile

- Die Qualität der Top-Arcade-Spiele hat längst die Qualität von Vollpreisspielen erreicht. Die grafische Messlatte liegt hoch. Und wer sie nicht überspringt, läuft Gefahr, nicht bemerkt zu werden.
- Durch das Interesse namhafter Publisher an der Plattform ist es generell schwierig, überhaupt auf die Plattform zu kommen, da die Slots im Jahr begrenzt sind.
- Die Aufmerksamkeitsspanne bschränkt sich auf die erste Woche – die Zeit bis zum nächsten Mittwoch. Wer in einem ungünstigen Umfeld erscheint, verliert wertvolle Aufmerksamkeit.
- Nur wenige Spiele entwickeln sich zum Selbstläufer und erreichen über einen längeren Zeitraum hinweg hohe Downloadzahlen, die mit einem »Bestsellerlisten-Effekt« weitere Downloads nach sich ziehen.
- Die direkte Konkurrenz zu anderen Spielen, die am gleichen Tag oder in der gleichen Woche erscheinen, wird immer entscheidender. Ein günstiges Timing kann wesentlich zum Erfolg beitragen.
- Die allgemeine Aufmerksamkeit für Arcade-Spiele ist geringer als die für Boxspiele: Häufig werden die Spiele nicht oder nur in geringem Umfang auf Websites und in den Magazinen besprochen.
- Die für das Approval von Microsoft notwendigen technischen Anforderungen sind insbesondere in puncto Online-Multiplayer, Stabilität und Error Handling nur von erfahrenen Programmierern erfüllbar.

Fazit

Letzen Endes ist es genauso schwer, riskant und aufwändig geworden, ein Spiel für XBLA zu finanzieren und zu entwickeln wie ein herkömmliches Konsolenspiel. Es sind nur andere Vor- und Nachteile, mit denen man konfrontiert wird. Unterm Strich sollte das Konzept ausschlaggebend sein: Wer ein Spiel wie Ion Assault entwickeln will, das auf die Plattform ausgelegt ist, hat kaum eine andere Chance, als sich um einen Platz auf Xbox Live Arcade oder PlayStation Network zu bemühen. Wer die Plattform primär als Ausweichmöglichkeit sieht, um den schwierigen Box-Markt zu umgehen, sollte sich genau mit den Besonderheiten der Plattform auseinandersetzen und ein Produkt entwickeln, das in der Lage ist, dort zu bestehen.

Dominik Mieth, Roman Pfrengle

Die einzelnen **Effekt-Layer** …

… aus denen eine **Explosion** zusammengesetzt wird. Die Timings der Frames für die einzelnen Ebenen werden vom Grafiker direkt im FX-Editor aufeinander abgestimmt.

Die zehn Stolpersteine Download-Spiele für Konsolen

Der digitale Spielevertrieb ist verführerisch einfach, birgt aber auch diverse Stolpersteine. Wir erläutern die zehn wichtigsten Dinge, die Entwickler beachten müssen, wenn sie sich für diesen Geschäftszweig entscheiden.

Im Hüpfspiel **Braid** (Xbox Live Arcade) können Sie die Zeit zurückdrehen. Dieses innovative Konzept hob den Titel von der Konkurrenz ab – ein wichtiges Alleinstellungsmerkmal und ein Kaufanreiz für Microsofts Kunden.

Pete Walentin

ist Development Director von keen games.

Bereits seit zehn Jahren arbeitet Pete Walentin in der Spielebranche. Er gründete keen games mit und sitzt darüber hinaus im Vorstand der gamearea Frankfurt Rhein Main e.V.

Wie das immer im Leben so ist, sind die seltensten Pfade, die man beschreitet, ohne Risiken. Nicht voranzuschreiten bedeutet aber im Umkehrschluss, sich jeglicher Chancen zu berauben. Die folgenden Informationen beleuchten die Risiken und Unwägbarkeiten bei der Entwicklung von Download-Spielen. Über diese sollte man sich im Klaren sein, um sich auch wirklich die Chancen zu wahren, die diese neue Form der digitalen Veröffentlichung bietet.

Die Plattformen

Microsoft startete im November 2004 Xbox Live Arcade (XBLA) auf der Xbox. Im November 2005 folgte XBLA auf der Xbox 360. Sony eröffnete im November 2006 zusammen mit der Markteinführung der PlayStation 3 das PlayStation Network (PSN), das ebenso über die PlayStation Portable zugänglich ist. Nintendo zog im März 2008 mit WiiWare nach, und DSiWare kam mit der Veröffentlichung des DSi in 2009. Nicht zu vergessen ist natürlich auch der App Store von Apple, der im Juli 2008 seine Pforten öffnete.

Auch abgesehen davon, dass die Xbox 360 stärker verbreitet ist als die PlayStation 3, ist es nicht weiter verwunderlich, dass Microsoft bei der digitalen Distribution eine Führungsrolle einnimmt. Schließlich hat der Hersteller als Erster das Segment der digitalen Distribution von kleineren Titeln für die großen Konsolen eingeführt, gefolgt von Sony und dem noch relativ frischen Wii- bzw. DSiWare-Shop.

Den App Store sowie die Entwicklung für das iPhone bzw. den iPod touch lasse ich in diesem Artikel außen vor. Die Hürden, eine App in den Store zu bekommen, sind verglichen mit denen für die großen Konsolen sehr niedrig. Zudem ist die Qualität der meisten Apps doch mehr als fragwürdig, was wiederum zu der interessanten Diskussion führt, ob die Plattformen gewissen Qualitätskontrollen unterliegen oder ob sie offen sein sollten. Ich denke, beides hat seine Vor- und Nachteile.

Konzentrieren wir uns aber jetzt auf den Kern des Themas: Was gibt es zu beachten, wenn ein Entwickler sich entscheidet, ein Download-Spiel für Konsolen zu entwickeln?

1. Die Registrierung

Wie es landläufig bekannt ist, benötigt man zum Entwickeln für eine spezifische Plattform auch die nötige Hardware. Die Entwicklungskits von Microsoft, Sony und Nintendo bekommt man nur, wenn man registrierter Ent-

wickler ist. Etablierte Studios sind dies im Allgemeinen auch, oder sollten zumindest keine Probleme haben, diesen Status zu erlangen. In erster Linie handelt es sich schließlich um einen formalen Prozess. Auch gänzlich neue Studios haben in der Regel keine Probleme, diesen Status zu erreichen. Allerdings müssen sie nachweisen, dass sie sich mit der jeweiligen Plattform auskennen. Voraussetzung ist zudem eine Gesellschaftsform, die den Plattformhaltern eine gewisse Sicherheit gibt – in Deutschland ist das etwa die GmbH.

2. Die Idee

Mal abgesehen davon, dass es sehr sinnvoll ist, mit etwas Neuem, Innovativem aufzuwarten, erhöht dies auch ungemein die Chancen bei den Plattformhaltern, eine Zustimmung für das Spiel zu erhalten. Denn ohne diese bleibt der angestrebte Markt unerreichbar – eines der Hauptrisiken in der Pre-Production-Phase.

Bei dieser ersten Hürde unterscheiden sich die Lizenzgeber deutlich. Während Nintendo ein relativ simples Verfahren einsetzt, ist dies bei Sony und Microsoft erheblich aufwändiger. Microsoft legt sehr großen Wert darauf, keine Spiele in das XBLA-Sortiment aufzunehmen, die es im Grunde schon gibt. Entsprechend sind Alleinstellungsmerkmale unbedingt vonnöten.

Man sollte den Plattformhalter als Publisher betrachten, dem man seine Idee verkaufen möchte. Sony hat ebenso ein sehr großes Interesse daran, dass die Spiele für ihre Plattformen genau diese Aspekte berücksichtigen. Auch wenn wir hier »nur« von Download-Spielen sprechen, steht Innovation – sowohl im technischen als auch im gestalterischen Bereich – ganz oben auf der Liste. Die Plattformhalter wollen unbedingt sicherstellen, ihren Kunden hochwertige und attraktive Spiele anzubieten. Vor allem wollen sie darüber hinaus auch Alleinstellungsmerkmale gegenüber den Spielen auf den Konkurrenz-Plattformen sehen.

3. Mit Publisher, ohne Publisher?

Da sich bei Download Games die Entwicklungskosten erheblich von denen der Boxed Products unterscheiden, ist es theoretisch möglich, solche Titel als unabhängiger Entwickler selbst zu finanzieren. Damit eröffnet sich die Option, die Rechte an der Marke (IP) zu behalten. Es bedeutet aber auch, das gesamte Risiko zu tragen und mit Aufgaben konfrontiert zu sein, an die anfangs vermutlich noch keiner gedacht hat.

Außerdem gib es einen weiteren Haken: Ohne Publisher kann man sein Spiel nur bei Nintendo und Sony veröffentlichen, nicht jedoch bei Microsoft. Es sei denn, man gewinnt den Xbox-Hersteller selbst als Publisher. Und je nachdem, wie viel investiert wird, kann man es sich eigentlich nicht leisten, mit der Xbox 360 die Plattform mit den besten Verkaufszahlen komplett außen vor zu lassen. Sollte man sich dennoch entscheiden, das Produkt ohne Publisher zu veröffentlichen, muss man sich vor Augen halten, welche zusätzlichen Aufgaben dies mit sich bringt – und damit auch Arbeit und Kosten. Da wären zum einen die Altersfreigaben für alle Regionen, in denen man veröffentlichen möchte. Dann die Lokalisierung, denn die Lizenzgeber fordern häufig mehrere Sprachversionen. Nicht vergessen darf man ebenso die (interne) Qualitätssicherung sowohl für das Spiel als auch für die Erfüllung der technischen Vorgaben, das Handhaben der Submission-Prozesse bei den Plattformhaltern, Markeneinträge sowie natürlich Marketing und PR. Dieser Aufwand ist nicht zu unterschätzen, und im Normalfall ist das entsprechende Know-how bei einem Entwickler nicht vorhanden.

Noch eine unverbrauchte Idee: Ähnlich wie im PC-Klassiker Lost Vikings von Blizzard kombinieren Sie im Puzzlespiel **Trine** (PlayStation 3) die Fähigkeiten mehrerer Helden, um vertrackte Levels zu meistern.

4. Online oder offline?

Da Download-Spiele online erworben werden, ist es natürlich naheliegend, in ein solches Spiel auch einen Online-Modus zu integrieren. Mal abgesehen davon, dass dies die Plattformhalter wünschen und es die Attraktivität des Titels vergrößert, erweitert es auch die Erfolgschancen. Zum einen ist ein Online-Multiplayer-Modus ein starkes Feature, zum anderen kann sich eine Community bilden und etablieren, wodurch die Halbwertzeit des Titels verlängert und damit auch der wirtschaftliche Erfolg gesteigert wird. Allerdings bedeutet das Integrieren eines Online-Multiplayer-Modus aber auch einen erheblichen Mehraufwand und erhöht dementsprechend die Kosten.

5. Downloadable Content

Ein weiterer Faktor, der für die Langlebigkeit des Titels und seines wirtschaftlichen Erfolges von Bedeutung ist, stellt Downloadable Content (DLC) dar. Es ist sinnvoll, sich von Anfang an darüber im Klaren zu sein, ob man DLC nachschieben möchte oder nicht. Wenn man DLC plant, sollte das Programm möglichst so ausgelegt sein, dass man nur Assets nachschieben

Keen Games

Das Frankfurter Entwicklerstudio hat unter anderem Anno: Erschaffe eine neue Welt (Wii und DS) sowie G-Force (PSP und DS) entwickelt. Derzeit arbeitet das Team unter anderem an zwei Download-Spielen für Konsole. Weitere Infos finden Sie unter www.keengames.com.

Im Multiplayer-Titel **Fat Princess** (PlayStation 3) prügeln sich zwei Spielerteams, um Nahrung für ihre übergewichtige Königstochter zu sammeln – einer der beliebtesten Downloads im PlayStation Network.

muss. Mit anderen Worten: Man sollte im Spiel die Bereiche, die eine Ergänzung oder Veränderung mit dem DLC erfordern, per Script steuern können. Das hat den Vorteil, dass DLC, der nur aus Assets besteht und keinen eigentlichen Code beinhaltet, welcher das schon vorhandene Spiel verändert, einem erheblich geringeren Testaufwand unterliegt und damit leichter durch die Qualitätssicherung der Plattformhalter geschleust werden kann.

6. Technische Vorgaben

Jeder Plattformhalter hat bestimmte technische Vorgaben, die bei der Entwicklung für das jeweilige System berücksichtigt werden müssen. Diese umfassen fast alle Bereiche des Spiels – etwa in welcher Sprache Buttons benannt sein müssen, wie sich das Programm in verschiedensten Situationen verhalten soll, bis hin zu den inhaltlichen Anforderungen wie Scoreboards, Leaderboards, Achievements, Trophies, Friendcodes, Manuals, Trial Versions, Shop im Spiel und noch vieles mehr.

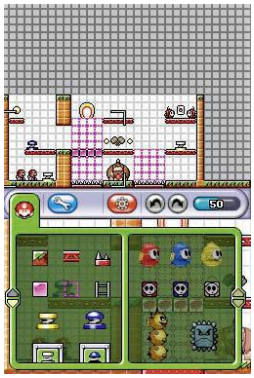
Das Einhalten der technischen Vorgaben der Plattformhalter ist ein sehr großes und weites Feld. Sollte der Entwickler für die eine oder andere Plattform schon ein Spiel designt und erfolgreich veröffentlicht haben, dann weiß er, was dies an Aufwand bedeutet und wie sehr dieser Faktor Einfluss auf den Veröffentlichungstermin haben kann. Wenn man sich nun entscheidet, ein Spiel für verschiedene Systeme zu entwickeln, potenziert sich der Aufwand, da natürlich jedes System seine eigenen Regeln hat. In einigen Bereichen überschneiden sich diese, aber es gibt auch viele Bereiche, die unterschiedlich gehandhabt werden müssen. Dies wiederum bedeutet, dass Code, Menüs etc. auf den einzelnen Plattformen nicht wiederverwendet werden können, sondern für jede Plattform einzeln entwickelt bzw. modifiziert werden müssen. Man sollte sich bereits zu Beginn der Entwicklung ausführlich mit diesen Vorgaben auseinandersetzen, weil es sonst zu einem sehr bösen Erwachen kommen kann.

7. Verkaufszahlen

Verkaufszahlen sind zwar jedes Mal nur eine Prognose, es ist aber dennoch sinnvoll, eine ungefähre Vorstellung davon zu haben, wie viele Einheiten das Spiel verkaufen könnte. Leider gibt es keine zuverlässigen Quellen, was Verkaufszahlen für Download-Spiele für Konsolen betrifft. Es existieren aber so einige Informati-

Simple Idee, hervorragend umgesetzt: Das Quizspiel **TV Show King** zählt zu den beliebtesten Downloads auf Nintendos Plattform WiiWare.

Genau genommen kein Spiel, trotzdem ein erfolgreiches WiiWare-Programm: Der Simulator **My Aquarium** macht den Flachbild-Fernseher zum Fischtank.



Der Markt für Download-Spiele boomt (neben dem iPhone) auch auf den wichtigsten Handheld-Konsolen, hier **Mario vs. Donkey Kong: Die Rückkehr der Mini-Marios** (links, DSi) sowie **Need for Speed: Shift** (PSP).

onen, die man nutzen kann, um sich ein Bild zu machen, sei es im Web oder in den Spielen selbst. Ein sehr guter Indikator sind zum Beispiel Leaderboards in Spielen. Ebenso interessant ist ein Vortrag von Simon Carless (Chefredakteur von Gamasutra), der auf der GDC 2009 ausführlich über Verkaufszahlen gesprochen hat – jedoch auch hier mit Ungenauigkeiten.

8. Zeitplanung

Generell gibt es bei der Spieleentwicklung viele unsichere Variablen. Bei Download-Spielen für Konsolen kommen jedoch noch weitere Risiken hinzu, die nicht nur die zeitliche Planung erschweren, sondern das ganze Projekt in Gefahr bringen. Deshalb ist es sehr ratsam, einen ordentlichen Puffer einzuplanen, um diese Risiken minimieren zu können.
Die wichtigsten Stolpersteine sind:

- Erhebliche Nachbesserungen in der Pre-Production am Design und damit auch am Umfang des Spiels, um es in einen Zustand zu bringen, der für den jeweiligen Plattformhalter »sexy« ist. Hier sind durchaus auch Spielinhalte vonnöten, die exklusiv für eine Plattform bestimmt sind.
- Nachbesserungen im Alpha-Stadium, da ein abschließendes Approval vom Plattformhalter noch aussteht.
- Der finale Approval-Prozess, der dem Überprüfen der technischen Vorgaben dient. Hier entsteht das Problem, dass jede nicht bewältigte Submission den Release des Titels um zwei bis vier Wochen verschieben kann.

9. Kostenkalkulation

Neben den klassischen Entwicklungskosten fallen natürlich die Kosten für die Devkits an, die zum Teil erheblich sind. Der QA-Aufwand, gerade für Spiele, die einen Online-Modus bieten, ist ebenfalls nicht zu unterschätzen. Dazu kommen die oben genannten, nicht entwicklertypischen Aufgaben wie Lokalisierung, QA der Lokalisierung, Defect Tracking, Pre-Certification QA, weltweite Age Ratings, Marketing (Trailer, Artworks, Websites, Foren), PR, Markeneinträge, Rechtsanwaltskosten etc.

Zu berücksichtigen ist natürlich, dass es vom Verkaufspreis eines jeden Titels zwischen Plattformhalter und Publisher/Entwickler einen Split gibt, sprich die Einnahmen geteilt werden müssen. Einige Plattformhalter stellen dem Entwickler sogar die Traffic-Kosten in Rechnung, die beim Download des Spiels anfallen.

10. Chancen

Die digitale Distribution hat schon jetzt einen erheblichen Anteil am Vertrieb von Software und demnach auch an Spielen. Dieser wird in den nächsten Jahren massiv steigen. Das liegt zum einen daran, dass zukünftig flächendeckende hohe Bandbreiten zur Verfügung stehen werden und die Kosten der digitalen Distribution gegenüber der herkömmlichen Distribution im Handel schon jetzt um ein Vielfaches geringer sind. Kleinere Download-Spiele sind zudem für ein Studio eine gute Möglichkeit, sich zu etablieren und auch mal andere Wege als die üblichen zu gehen.

Im Bereich der Download-Spiele auf Konsole gibt es noch sehr viel Potenzial. Um dieses zu nutzen, sollte man sich der genannten Risiken und Unwägbarkeiten bewusst sein und so früh wie möglich gegensteuern. **Pete Walentin**

Checkliste

- ✔ **Innovative Spielidee**: sollte auf den Ziel-Plattformen keinesfalls bereits in dieser oder einer ähnlichen Form existieren.
- ✔ **Marktanalyse**: wer ist die Konkurrenz und worin hebt sich die eigene Idee ab; was sind die Alleinstellungsmerkmale; wie sind die Verkaufszahlen der Konkurrenz?
- ✔ **Zeitplanung**: sollte sehr detailliert ausfallen, mit großzügigen Puffern für Nachbesserungen und die finale Freigabe.
- ✔ **Kostenaufstellung**: Entwicklungsbudget, Publishing-Kosten und Traffic-Kosten den prognostizierten Einnahmen gegenüberstellen; eine originelle Idee, die sich nicht rechnet, ist eine schlechte Idee.
- ✔ **Plattformhalter**: so früh wie möglich mit den Plattformhaltern in Kontakt treten und die nötigen Konzepte einreichen.
- ✔ **Technische Vorgaben**: schon in der Design- und Planungsphase mit den Vorgaben der Plattformhalter auseinandersetzen und diese während der Entwicklung immer berücksichtigen, um Korrekturen zu vermeiden.
- ✔ ... and last but not least; ein tolles Spiel entwickeln, viel Spaß dabei haben, nicht wahnsinnig und vor allem ganz am Ende reich und berühmt werden.

Ein Online-Modus kann die Lebenserwartung eines Download-Programms deutlich erhöhen und für eine treue Fangemeinde sorgen, hier etwa ein Mehrspieler-Gefecht in **Atomic Bomberman** (Xbox Live Arcade).

»Wie bei einem Rockkonzert« Im Gespräch mit Dan Connors

Telltale Games ist derzeit wohl das erfolgreichste, auf digitale Distribution und Episodenspiele spezialisierte Studio. CEO Dan Connors über die alten Zeiten, die aktuellen Herausforderungen und die Chancen der Zukunft.

Dan Connors

ist Gründer und Geschäftsführer von Telltale Games.

Dan Connors begann seine Karriere 1993 bei LucasArts. 2004 gründete er gemeinsam mit Kevin Bruner das Entwicklerstudio Telltale Games, mit dem er sich von Anfang an auf digital vertriebene Episodenspiele spezialisierte. Inzwischen hat sein Unternehmen fast 30 Titel veröffentlicht. Zu den erfolgreichsten Marken zählen die Reihen Sam & Max, Wallace & Gromit und die Fortsetzung der Monkey-Island-Serie.

WWW
makinggames.de

- Liquid-Chef Ed Del Castillo über die Herausforderungen als unabhängiges Entwicklerstudio

Making Games War es bei der Geschäftsgründung von Anfang an klar, dass ihr euch auf digitale Distribution spezialisieren wollt?
Dan Connors Oh ja, das war damals sogar eine der Hauptentscheidungen bei der Gründung der Firma. Wir registrierten den Aufschwung der digitalen Distribution und uns war klar, dass wir damit Zielgruppen ansprechen konnten, die von den klassischen Retail-Kanälen sozusagen unterbeliefert wurden.

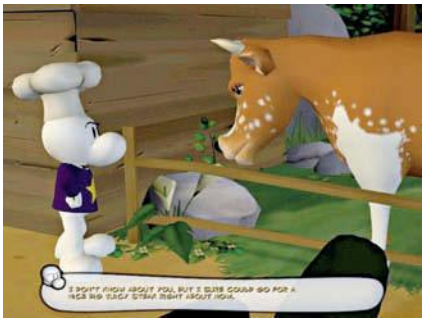
Making Games Welche Zielgruppen wurden denn damals vernachlässigt?
Dan Connors Die für storylastige Spiele, die für Casual Games – also quasi alle, die sich nicht für Schießen, Autos oder Sport interessieren (lacht). Es gab da eine Nachfrage, die von den Publishern nicht bedient wurde. Wir dachten uns, dass wir einspringen und diese Nachfrage bedienen könnten, und daraus wollten wir ein Geschäftsmodell aufbauen. Außerdem haben wir mit der digitalen Distribution eine Menge Probleme bei der Entwicklung, beim Pitching und so weiter gelöst. Wir sahen ganz einfach die Möglichkeit, auf diese Art die Leute zu erreichen, die sich genau für die Spiele interessieren, die wir entwickeln wollen. Natürlich mussten wir dabei auch viel experimentieren. Und all das war eben nur über den Weg der digitalen Distribution möglich.

Die Telltale-Homepage: »Wir mussten ein Web-Team aufbauen, eine E-Commerce-Lösung finden und uns auch mit Marketing und Publishing auseinandersetzen.«

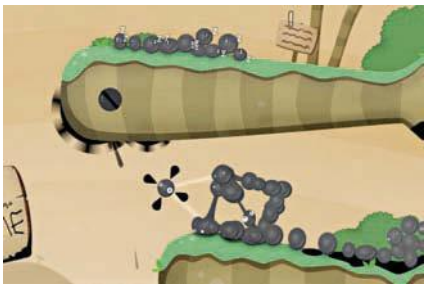
Making Games Wo du gerade davon sprichst: Glaubst du, dass sich die Zielgruppen von Download- und Retail-Spielen unterscheiden?
Dan Connors Nicht unbedingt. Die Erlebnisse sind allerdings unterschiedlich. Also muss man wissen, auf welche Erfahrungen die Leute reagieren, um digitale Distribution zu bevorzugen. Es arbeitet ein ganzer Haufen an Leuten daran, das zu definieren. Nicht nur Telltale, auch Xbox Live Arcade, WiiWare oder Steam. All diese Leute versuchen zusammen mit den Kunden herauszufinden, wie die Benutzererfahrung mit digitaler Distribution sein sollte. Da gibt es noch viel zu entdecken, und das ist großartig.

Making Games Musstet ihr eure Firma anders aufbauen, als wenn ihr euch für den klassischen Retail-Verkauf entschieden hättet?
Dan Connors Definitiv! Zum Beispiel mussten wir das Produkt richtig hinbekommen. Das Studio selbst hat zwar sehr traditionelle Strukturen, aber die Werkzeuge und Technologien, die wir im Prozess einsetzen, sind vor allem auf schnelle Wiederholungen und schnelle Entwicklung von Episoden ausgelegt. Wir haben das Studio also mit der Kapazität aufgebaut, möglichst viele Inhalte und vor allem auch möglichst viele Iterationen generieren zu können. Schnell und kontinuierlich die richtige Mischung von unterschiedlichen Produkten zu finden. Das ist ein Teil der Vision. Der zweite Teil der Vision ist, so etwas wie ein Fernsehstudio zu schaffen, im Gegensatz zu einem Filmstudio. So kann man von seinem Investment in ein Franchise profitieren. Also eine Marke schaffen und daraus wiederum eine anhaltende und regelmäßige Erfahrung bauen. Und wir wollten das alles als unabhängiger Publisher bewerkstelligen. Dazu mussten wir ein Web-Team aufbauen, eine E-Commerce-Lösung finden und uns auch mit Marketing und Publishing auseinandersetzen. Aber wir haben grundsätzlich immer mit dem Produkt begonnen. Wir verwendeten das Produkt dazu, einen Kontakt mit der Zielgruppe zu initiieren. Und

»Microsoft weiß, was nötig ist, um einem Produkt ausreichend Aufmerksamkeit zu geben, sodass es genügend Schwung bekommt, um auf der Plattform erfolgreich zu sein.«



Das erste Telltale-Spiel Bone: »Wir registrierten den Aufschwung der digitalen Distribution und uns war klar, dass wir damit Zielgruppen ansprechen konnten, die von den Retail-Kanälen unterbeliefert wurden.«



»Auf jedes World of Goo kommen Hunderte Spiele, die es eben nicht geschafft haben.«

sobald dieser Teil der Zielgruppe nicht mehr gewachsen ist, haben wir Marketingtechniken, andere Distributionswege und sonstige Business-Möglichkeiten genutzt, und dazu gehört übrigens auch der von dir erwähnte klassische Retail-Verkauf.

Making Games Wie wichtig ist das Marketing bei digitaler Distribution?
Dan Connors Ich denke, Marketing ist immer dann wichtig, wenn man ein Publisher sein will. Jeder Publisher muss Marketing betreiben. Eine Online-Marketingkampagne, die zum Online-Kauf eines Spiels führt, ist anders als eine Kampagne, die einen Käufer dazu bringt, in den Laden zu gehen. Es ist ein anderes Skillset, ein anderes Verständnis, ein anderer Aufbau. Und diese ganzen Prozesse kann man eigentlich nur wirklich verstehen, wenn man möglichst viel praktische Erfahrung sammelt. Jedes Produkt, das wir auf den Markt bringen, hilft uns also beim Lernen – glücklicherweise sind es inzwischen schon 29 (lacht).

Making Games Und was ist mit Entwicklern, die auf digitale Distribution setzen? Wie viel Marketing brauchen die?
Dan Connors Das kommt darauf an, ob sie auch selbst veröffentlichen wollen. Wenn man nämlich jemanden für das Publishing bezahlt, bezahlt man damit auch für das Marketing. Man bezahlt für die Distributionsbeziehungen und für das Management der Konsolen-Kanäle. Dein Job ist es zu entwickeln. Wenn man also ein Entwickler sein will und das alles ist, worum man sich kümmern möchte, dann sollte sich auch ein Publisher um all die anderen Dinge kümmern.

Making Games Ich frage, weil es für viele unabhängige Entwickler sehr verlockend klingt, um traditionelle Publisher einen Bogen zu machen und ihr eigenes Ding durchzuziehen.
Dan Connors Ja, aber wenn man selbst veröffentlichen will, muss man in diese Richtung auch investieren. Das muss man sich bewusst machen, egal wie verführerisch die selbstständige Online-Distribution zunächst klingt.

Making Games Ich glaube, dass viele Entwickler auf virales Marketing setzen. Sie glauben, wenn sie ein gutes Spiel entwickeln, wird es sich schon herumsprechen.
Dan Connors (lacht) Ja, das stimmt. Und manchmal passiert das ja wirklich. Man weiß einfach nicht, wie sich die Wirtschaft entwickeln wird. Aber es gibt immer irgendwo die Distribution. Und wer die Transaktionen verwaltet, verdient das Geld – außer das Spiel ist kostenlos. Wenn du zum Beispiel etwas wie World of Goo machst, was ja erfolgreich war, und versuchst, es so vielen Leuten wie möglich zu verkaufen, musst du irgendwann mit einem Partner zusammenkommen, der das Projekt finanziert. Außer du hast selbst genug Geld.

Making Games Aber kann man sich nicht einfach mit einer Distributionsplattform wie Steam oder GamersGate zusammentun?
Dan Connors Ja, aber auch diese Plattformen wollen Geld verdienen, und dein Spiel hat ja auch nur eine begrenzte Lebensdauer. Und wenn deine zehn Jungs ein Projekt zusammenbauen wollen, das ein paar Millionen Dollar Umsatz macht, dann wird virales Marketing reichen. Aber wenn man eine 50 oder 100 Millionen Dollar Firma aufbauen will, muss man sehr viel mehr Arbeit hineinstecken, um Profit zu erzielen. Mit viralem Marketing bekommt man sozusagen viralen Erfolg. Es sieht von außen immer besser aus als aus wirtschaftlicher Sicht. Wenn es keine vernünftige Struktur zur Monetarisierung gibt, werden eine ganze Menge kostenlose Kopien im Umlauf sein und alle möglichen Partner werden einen Teil des Gewinns abbekommen. Auch wenn es nur ein 10-Dollar-Produkt ist – wenn man es eine Million Mal ver-

» Es gab eine Nachfrage, die von den Publishern nicht bedient wurde. «

» Wir verwenden das Produkt dazu, einen Kontakt mit der Zielgruppe zu initiieren. «

» Marketing ist wichtig, wenn man ein Publisher sein will. «

»Distributionspartner haben ihre eigenen Möglichkeiten und ihre eigenen Pläne. Und diese Pläne stimmen nicht immer mit denen der Entwickler überein.«

kauft, bekommt man zehn Millionen Dollar. Beim Entwickler kommen davon aber vielleicht nur fünf Millionen an. Was immer noch eine Menge Geld ist, doch was soll man als Nächstes damit tun? Hört man einfach auf und geht in Rente oder reinvestiert man und baut ein Geschäft auf? Spätestens dann muss man sich Gedanken über einen soliden Geschäftsplan machen. Denn du wirst nicht mit jedem Spiel solch einen Erfolg haben. Auf jedes World of Goo kommen Hunderte Spiele, die es eben nicht geschafft haben.

> Mit viralem Marketing bekommt man sozusagen viralen Erfolg.

Making Games Worauf kommt es an, wenn man seinen Fuß in die Türen der großen Konsolen-Plattformen bekommen möchte? Wie schwierig war es für euch?
Dan Connors Das ist kompliziert. Entweder ist es technisch herausfordernd, oder aber der Plattformpartner ist vom Programmieren her herausfordernd. Etwas für den PC aufzubereiten, ist sehr viel einfacher. Aber auch da ist es schwierig, mit den Distributionspartnern zusammenzuarbeiten. Die haben ihre eigenen Möglichkeiten und ihre eigenen Pläne. Und diese Pläne stimmen nicht immer mit denen der Entwickler überein.

> Publishing ist ein Geschäft. Und die Entwicklung ist der kreative Teil des Geschäfts.

Making Games Was sind denn die größten Herausforderungen aus technischer Sicht?
Dan Connors Wenn man für Konsolen entwickelt, muss man auf all die gesetzlichen Überprüfungen, Lokalisierungen und Achievements achten. Und dann gibt es da noch die vorgegebenen Meilensteine und Hardwarelimits, um die man sich kümmern muss.

Making Games Wie wichtig ist das Timing bei der digitalen Distribution? Microsoft bringt zum Beispiel nur zwei Spiele die Woche heraus.
Dan Connors (lacht) Bei unseren Episoden-Spielen ist das sicher nicht ganz so kompliziert. Aber die Verkaufskanäle können natürlich auch als Unterstützung gesehen werden. Microsoft weiß schließlich am besten, wie sie Einheiten verkaufen. Sie wissen, was nötig ist, um einem Produkt ausreichend Aufmerksamkeit zu geben, sodass es genügend Schwung bekommt, um auf der Plattform erfolgreich zu sein. Es stehen also gute Gründe hinter ihrer Entscheidung. Und sie sind da offensichtlich sehr umsichtig. Sie versuchen, bei jedem XBLA-Spiel so viel wie möglich zu verkaufen, was ja das Beste für alle ist. Aber es ist kein freier Vertriebskanal. Man muss definitiv mit einem Partner zusammenarbeiten, der ebenfalls etwas erreichen will.

Making Games Woran muss ich denken, wenn ich in die digitale Distribution einsteigen will? Es geht jetzt weniger um Millionen-Projekte, sondern eher um ein kleineres Produkt, das ich mit einem eher begrenzten Budget umsetzen will.
Dan Connors Die erste Frage, die man sich stellen sollte, lautet: »Warum wird das Spiel gekauft«? Wenn die Antwort dazu ist, weil es großartig wird, dann heißt es einfach »Okay, viel Glück«. Es ist gut, überzeugt zu sein, aber es steckt auch mit einem guten Produkt noch eine Menge harte Arbeit dahinter, es bekannt zu machen. Je mehr Augen auf das Produkt gerichtet sind, desto mehr Geld verdient man. Wenn man diese Augen von jemand anderem bekommt, musst du den auch bezahlen. Und das ist die entscheidende Frage: Wie groß muss dein Stück vom Kuchen sein, damit du auch satt wirst?

Making Games Es steckt also eine Menge Geschäftliches hinter der Produktion eines Spiels?
Dan Connors (lacht) Ja, deswegen nennt man es ja Geschäft. Publishing ist ein Geschäft. Und die Entwicklung ist der kreative Teil des Geschäfts. Wenn man nur kreativ arbeiten will, übergibt man dem Publisher die Verantwortung. Du machst etwas Großartiges, handelst einen guten Preis mit ihnen aus und kannst so weitermachen. Der Nachteil ist natürlich, dass du vom Publisher abhängig bist. Wenn der Publisher sagt, dein Produkt funktioniert für uns nicht, hast du ein mächtiges Problem.

Making Games Worauf muss ich beim Projektmanagement achten? Müssen meine Leute bei digitaler Distribution prinzipiell schneller oder fokussierter arbeiten?
Dan Connors Die Rechnung ist eigentlich ganz einfach: Der Ertrag der Investition sollte die Entwicklung vorantreiben. Das gibt vor, wie viele Leute man hat und wie lange man an dem Produkt arbeiten kann. Wenn man etwas erstellt, das keinen ausreichend hohen Ertrag verspricht, muss man alles noch mal überdenken. Und wahrscheinlich wird alles kleiner werden. Es ist einfach ganz anders, wenn man 300.000 Dollar bekommt, als wenn man 10 Millionen Dollar für ein Retail-Produkt bekommt. Dann sind es eben nicht mehr 80 Leute, sondern 10.

Making Games Was denkst du über die gewachsene Internet-Bandbreite? Man kann inzwi-

schen komplette Xbox-360-Titel herunterladen. Ist das eine Konkurrenz für die typischen, kleinen Download-Spiele?
Dan Connors Ich glaube nicht so sehr. Die traditionellen Spiele tun das, was sie am besten können. Die Download-Spiele haben hier eine Möglichkeit, etwas anderes zu bieten. Eine Erfahrung, die es vorher nicht gab. Retail hat die Spiele limitiert. Es gibt Vorgaben, was ein Spiel sein kann. Wir hätten im Handel doch niemals sechs Spiele in sechs Monaten in die Regale stellen können. Erst die digitale Distribution hat uns das ermöglicht. Sie hat auch Dinge wie mehr Benutzer-Einbeziehung, ständige Updates und Mikro-Transaktionen ermöglicht. All diese verschiedenen Wege sind konsistenter als der traditionelle Handel.

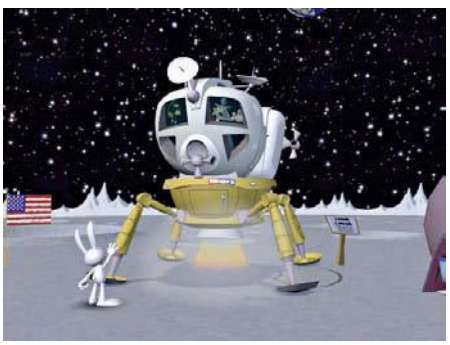
Making Games Was ist deiner Meinung nach eine geeignete Team-Größe, um in die digitale Distribution einzusteigen?
Dan Connors Das kommt ganz auf das Projekt an: Was möchte man machen und wem möchte man das Spiel verkaufen? Wie muss es aussehen und funktionieren, um die prognostizierten Verkaufszahlen zu erreichen? Und wie kommt das Spiel an diesen Punkt? Solche Fragen muss man als Allererstes und so ehrlich und präzise wie möglich klären. Die Antworten definieren dann, wie viel man in sein Team investieren kann. Aber unbedingt defensiv planen! Wenn man glaubt, eine Million zu verdienen, sollte man für die Entwicklung maximal 300.000 einplanen und ein Team anstellen, das 50.000 im Monat kostet. Wenn man mehr Gewinn erwartet, kann man auch ein größeres Team engagieren. Aber bevor man irgendwelche Entscheidungen trifft, sollte man die Größe seiner Zielgruppe herausfinden. Und dann natürlich, welche konkreten Inhalte sie von deinem Spiel erwarten.

Making Games Letzte Frage: Was ist aus eurer Sicht der größte Vorteil der digitalen Distribution gegenüber dem Ladenverkauf?
Dan Connors Ganz klar: konstantes Feedback aus der Zielgruppe. Die vielen Iterationen, mit denen wir uns monatlich verbessern können. Und wir haben richtigen Kontakt zur Zielgruppe. Ich vergleiche das manchmal mit Livemusik. Das Produkt ist live. Und wir und die Zielgruppe sind während der Entwicklung miteinander verbunden. Das motiviert das Team, beeinflusst Entscheidungen, und während wir weiterentwickeln, bekommen wir live das Feedback von unseren Fans. Es ist wie bei einem Rockkonzert, und wir sind die Musiker auf der Bühne. Was kann es Schöneres geben?

Interview: Heiko Klinge
Übersetzung: Sara Geller

> Die Download-Spiele haben die Möglichkeit, etwas anderes zu bieten.

> Bevor man irgendwelche Entscheidungen trifft, sollte man die Größe seiner Zielgruppe herausfinden.



Die erste Staffel von Sam & Max: »Wir hätten im Handel doch niemals sechs Spiele in sechs Monaten in die Regale stellen können.«

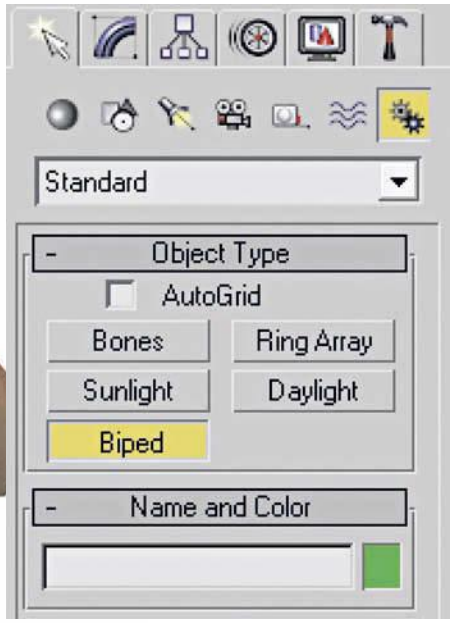


Abbildung 1: Biped Objekt in 3D Studio Max.

Character Design: Workshop Teil 4
Entwicklung und Design eines Spielecharakters

Im abschließenden Teil der Workshop-Reihe erweitern wir unser Charaktermodell »BadAss Rabbit« um erste Animationen.

Chris Noeth

ist Art Director von Rough Sea Games.

Der Diplominformatiker Chris Noeth arbeitet, nach langjähriger Tätigkeit im Bereich Entertainment-Illustration, seit 1999 als 2D-/3D-Artist in der Spielebranche und unterrichtete lange Zeit an der Fachhochschule Würzburg-Schweinfurt zum Thema »3D-Modellierung und Animation« Informatik- und Designstudenten. Zu seinen Projekten zählen neben PC-Vollpreisspielen auch über 30 Mobile Games. Darüber hinaus ist er Comiczeichner der US-Comicserien »Tales from the Crypt«, »Ray Harryhausen Presents« und seiner eigenen Serie »MAYA«, die er in Kürze als Webcomic veröffentlicht. Seit Anfang 2009 arbeitet er als Art Director für die Rough Sea Games GmbH (www.rough-sea.com).
Mehr zu Chris Noeth finden Sie auf:
www.chrisnoeth.com
Sie erreichen ihn per E-Mail unter:
chris@chrisnoeth.com

Wie bereits in den vergangenen Workshop-Teilen arbeiten wir auch im letzten Teil der »BadAss Rabbit«-Reihe mit 3D Studio Max und erstellen die Animation unseres Charakters mit dem Plug-in »Character Studio«. In den neueren Versionen von 3D Studio Max ist Character Studio automatisch integriert. Bei älteren Versionen wird Character Studio allerdings als Zusatz-Plug-in angeboten.

Der Workshop setzt voraus, dass Ihnen die Grundlagen der 3D-Animation bekannt sind. Sie sollten bereits eigene Keyframe-Animationen erstellt haben und den Umgang mit den wichtigsten Animationstools in 3D Studio Max kennen.

3D Studio Max bietet viele Möglichkeiten zur Animationserstellung. Wir arbeiten in diesem Workshop mit dem 3D Studio Max Biped von Character Studio, da dieses in der Spielentwicklung sehr beliebt ist und in der Praxis häufig angewandt wird. Bei einem Biped handelt es sich um ein vorgefertigtes, dem Menschen nachempfundenes Knochenskelett, welches beliebig an ein Charaktermodell angepasst werden kann. Durch die Verwendung eines Bipeds spart man sich das Erstellen einer komplexen Skelettstruktur, da ein Biped-Skelett lediglich durch einen Mausklick im Viewport erstellt wird. Ein weiterer Vorteil der Verwendung von Bipeds bei Charakteranimationen ist die Möglichkeit, vorgefertigte Motion Capture Files oder abgespeicherte Biped-Animationen auf jedes beliebig angepasste Biped zu übertragen. Nicht unerwähnt bleiben soll auch die Schrittanimation der Bipeds, mit der man einfach Fußspuren auf dem virtuellen Boden verteilt, denen das Biped-Skelett anschließend folgt. Man spart sich somit auch das Anlegen einer kompletten Keyframe-Animation von Hand. Die Eigenschaften von Character Studio ermöglichen ein sekundenschnelles Erstellen eines Charakterskeletts mit Animation. Sie müssen danach lediglich das Skelett an Ihr Modell anpassen, die Animation am Ende überarbeiten und sie anschließend nach Ihren Wünschen editieren. Sind diese Schritte abgeschlossen, können Sie das Modell in einem Format exportieren, welches von der entsprechenden Spiel-Engine automatisch erkannt und interpretiert wird.

Das Biped

Um in 3D Studio Max ein vorgefertigtes Biped-Charakterskelett zu erstellen, müssen Sie, wie in

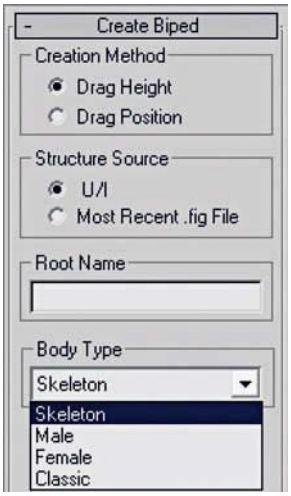


Abbildung 2 (links)**:** Biped Rollout.
Abbildung 3 (rechts)**:** Auswahl der Biped Körperart.

Abbildung 4: Darstellung der Biped Körperarten.

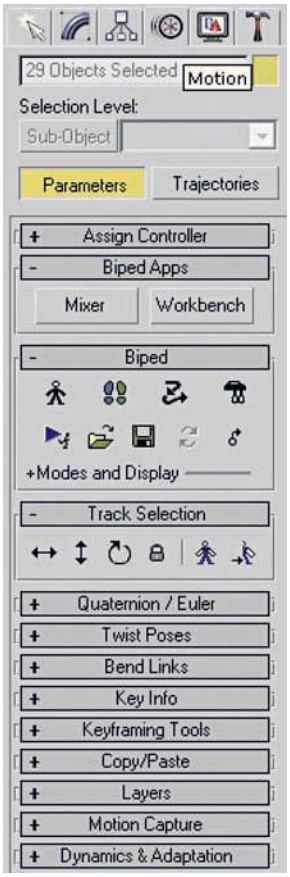


Abbildung 5: Biped Parameter.

Abbildung 1 gezeigt, in der Registeranzeige den Bereich »Systems« ganz rechts anwählen und danach in den Auswahlmöglichkeiten »Biped« aktivieren.

Abbildung 2 zeigt die Eigenschaften des Bipeds, das Sie durch einen Mausklick im Viewport erstellen können. Die wichtigsten Einstellungen für Bipeds sind die Details des Skelettsystems, mit denen Sie das Skelett genau auf Ihr vorhandenes Charaktermodell oder Ihre Kreatur abstimmen können. Über die Auswahl des Körpertyps (Body Type) können Sie zwischen den Biped-Modellen (Skeleton, Male, Female und Classic) auswählen. **Abbildung 3** zeigt die Auswahlmöglichkeiten und in **Abbildung 4** sehen Sie die unterschiedlichen Modelle in der genannten Reihenfolge von links nach rechts. Laden Sie vor dem Erstellen des Bipeds Ihr Charaktermodell, richten Sie dieses am Nullpunkt aus und erstellen Sie dann das Biped direkt neben dem Modell. Das Biped sollte die gleiche Höhe wie das Modell besitzen. Genauer: Die wichtigsten Körperteile des Bipeds sollten möglichst exakt den Größenverhältnissen des Modells entsprechen. **Abbildung 5** zeigt die Biped-Parameter. Passen Sie im nächsten Schritt das Biped an das Charaktermodell an. Damit die Anpassung übersichtlicher wird, sollten Sie den Charakter transparent machen. Die Transparenz können Sie über das »Quadmenu« einstellen. Das erreichen Sie mittels eines Rechtsklicks der Maus im Viewport von 3D Studio Max. Für die Transparenzdarstellung bietet sich das Fenster für die perspektivische Ansicht an, da Sie das Modell beim Anpassen der Envelopes vom Skin-Modifikator später ständig drehen und von allen Seiten betrachten werden. Deshalb sollten Sie den Rechtsklick im Perspektive-Fenster ausführen. **Abbildung 6** zeigt den Bereich des Quadmenus, über den Sie die Objekteigenschaften (Object Properties) erreichen können. Nun sollten Sie das Fenster aus **Abbildung 7** sehen, wo Sie unter anderem das ausgewählte Objekt transparent

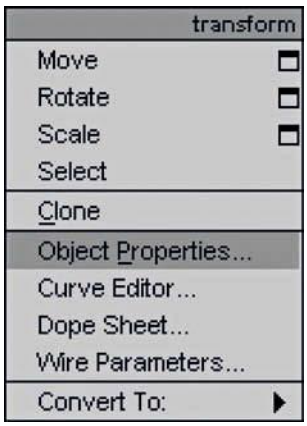


Abbildung 6: Objekteigenschaften im Quadmenu.

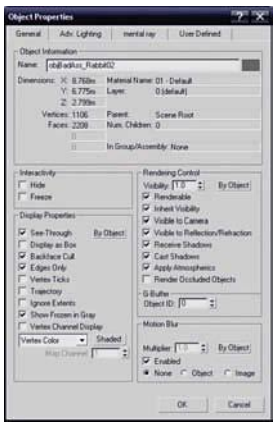


Abbildung 7: Durchsichtiges Darstellen über Objekteigenschaften.

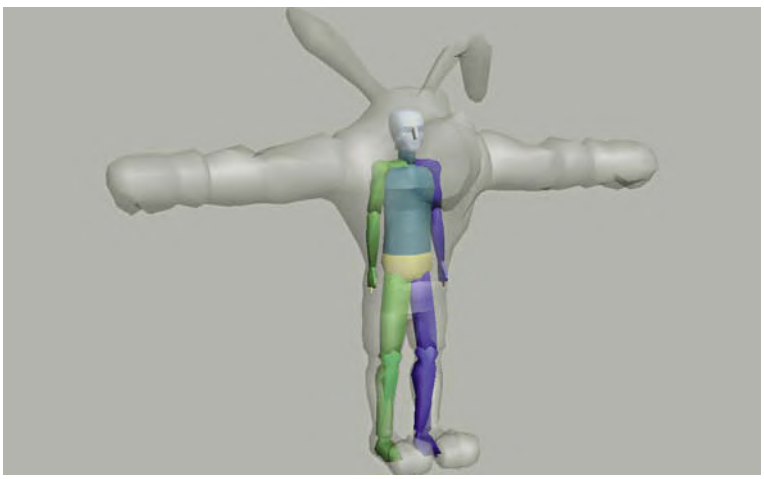


Abbildung 8: Transparente Darstellung des BadAss Rabbits.

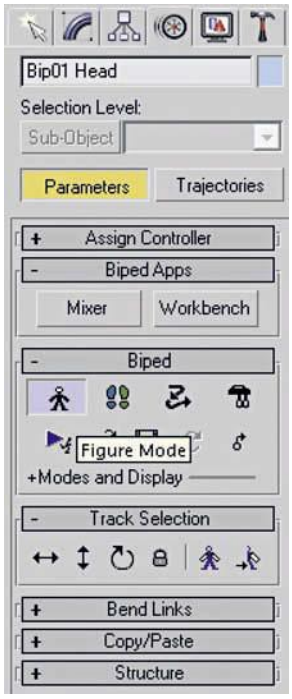


Abbildung 9: Aktivieren des Figurenmodus für Bipeds.

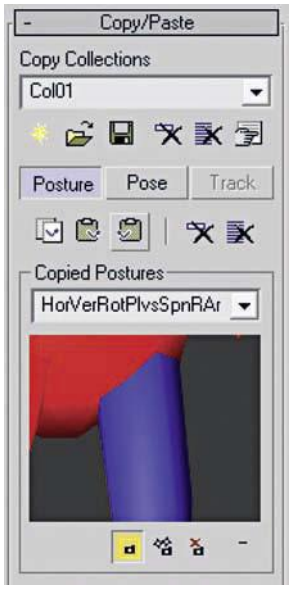


Abbildung 11: Spiegeln von Arm- und Beinposen beim Biped.

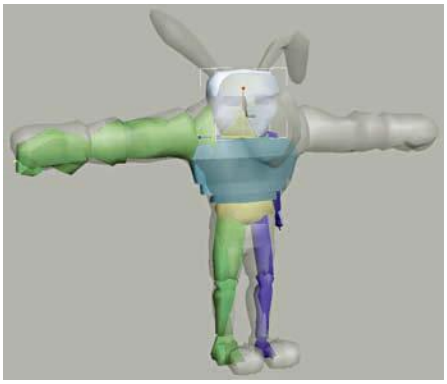


Abbildung 10: Einpassung an Charakter-Mesh.

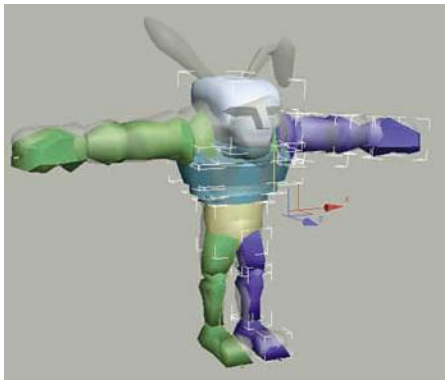


Abbildung 12: Gespiegelte Arm- und Beinpose.

schalten können. In Abbildung 8 sehen Sie schließlich das transparente Modell mit dem am Nullpunkt ausgerichteten Biped. Dies ist unsere Ausgangsbasis für die Anpassung der Biped-Teile an das 3D-Modell des Charakters.

Biped-Anpassungen

Um mit Hilfe des Bipeds das Charakter-Mesh zu beeinflussen, können Sie entweder mit dem Modifikator »Physique« oder dem Modifikator »Skin« arbeiten. Wir arbeiten im Workshop mit dem Modifikator Skin, da dieser aufgrund seines breiten Einsatzes in der Vergangenheit kontinuierlich verbessert wurde.

Aktivieren Sie für die Biped-Anpassung in den Biped-Eigenschaften den Button »Figure Mode« (Figurenmodus). Das Aktivieren dieses Buttons ermöglicht die Beeinflussung des Bipeds. Sie können nun mit Hilfe der Transformationswerkzeuge die einzelnen Teile des Skeletts an die Körperform des Hasen-Charakters anpassen. Den Button für den Figurenmodus in Form eines kleinen Männchens mit hellem Kopf können Sie in Abbildung 9 sehen.

Da das Charaktermodell symmetrisch erstellt wurde, können Sie die folgenden Schritte, die Sie für eine Hälfte des Skeletts anwenden, durch einen Klick in den Eigenschaften-Rollout des Bipeds unter der Gruppe »Copy/Paste« einfach auf die andere Hälfte übertragen.

Beginnen Sie nun mit der Anpassung der einzelnen Teile. Dieser Schritt ist wichtig, da die möglichst genaue Übereinstimmung des Skeletts mit der Form des 3D-Modells die spätere Bearbeitung der Einflussbereiche des Modifikators Skin deutlich vereinfacht. Der Grund dafür ist, dass sich die Einflussbereiche der einzelnen Bones, die später das Mesh des Charakters beeinflussen, an der Form der Bones orientieren. Je besser Sie also die Bones im Vorfeld am Mesh ausgerichtet haben, desto weniger müssen Sie später die Einflussbereiche kleinteilig korrigieren und bearbeiten. Abbildung 10 zeigt das zur Hälfte eingepasste Biped. In Abbildung 11 sehen Sie den Eigenschaftsbereich, mit dem Sie die Pose der eingepassten Hälfte des Bipeds auf die andere Hälfte des Charakters übertragen können. Das schließlich komplett an den Hasen angepasste Biped sehen Sie in Abbildung 12.

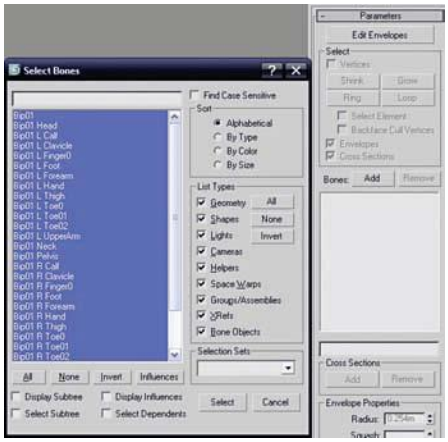


Abbildung 13: Gewählte Bones für den Skin-Modifikator.

Der Skin-Modifikator

Wählen Sie nach der Anpassung des Bipeds das Mesh des Charakters aus und fügen Sie anschließend den Modifikator Skin hinzu. Im Eigenschaften-Rollout des Skin-Modifikators finden Sie die Möglichkeit, die Bones zuzuweisen, die das Mesh beeinflussen sollen. Fügen Sie hier die Bestandteile des Bipeds hinzu, wie Ihnen Abbildung 13 zeigt. Die zugewiesenen Bones erscheinen daraufhin in der Liste des Modifikators Skin – das sehen Sie in Abbildung 14. Von hier können sie ausgewählt werden, um etwa die Wirkungsbereiche festzulegen. Falls Sie zuvor das Biped bereits gut an das Mesh anpassen konnten, befindet sich der Charakter aktuell in einem Zustand, in dem er schon grob animiert werden kann, ohne dass einzelne Punkte des Meshs unbeeinflusst bleiben.

Bereits zu diesem Zeitpunkt sollten Sie eine Animation zuweisen, damit Sie die schon aktiven Envelopes und deren Einwirkung auf das Charakter-Mesh testen und optimieren können. Sie können Ihre erstellten Animationen für ein Biped über den Button in Form einer Diskette im Eigenschaften-Rollout des Bipeds abspeichern. Auf diese Weise gesicherte Animationen können dann zu jeder Zeit jedem beliebigen Biped zugewiesen werden. Im Lieferumfang von 3D Studio Max sind bereits einige gute Animationen enthalten, die sich hervorragend zum Testen der Envelopes eignen.

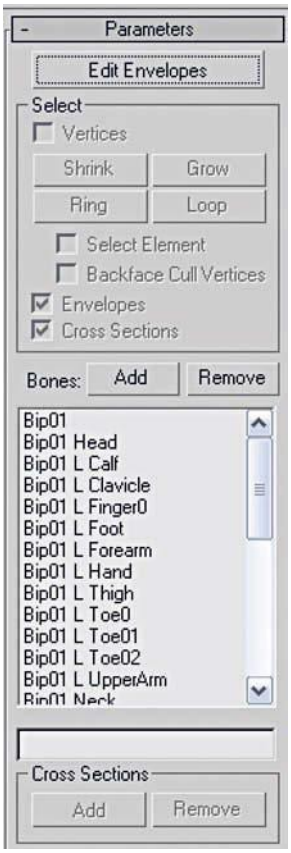


Abbildung 14: Hinzugefügte Bones in der Auswahlliste vom Skin-Modifikator.

Das Ghosting

Bevor Sie die Animation zuweisen, schauen wir uns noch kurz die Möglichkeit der Ghosting-Funktion an. Das Ghosting wird im Ansicht-Menü aktiviert, und Details der Darstellung können über die Eigenschaften im Viewport-Register von 3D Studio Max eingestellt werden (**Abbildung 15**). Um das Ghosting zu testen und später die Wirkungsbereiche der Envelopes des Skin-Modifikators festzulegen, deaktivieren Sie nun den Figurenmodus und laden eine vorgefertigte Animation über den Button »Animation laden« rechts unter dem Figurenmodus-Button. Sie können nun die Animationstools verwenden und die Animation abspielen. Mit aktiviertem Ghosting sollte das entsprechende Viewport-Fenster wie in **Abbildung 16** aussehen. In dieser Abbildung sehen Sie bei genauer Betrachtung noch einige Verzerrungen im Charakter-Mesh, und zwar am deutlichsten bei der grauen Pose im Bereich der Füße. Diese Verzerrungen werden nun bei der Bearbeitung der Envelopes des Skin-Modifikators genauer untersucht. Die Wirkungsbereiche werden so optimiert, dass es möglichst in keiner Pose Verzerrungen gibt und der Körper beziehungsweise die Proportionen des Modells wie im ersten 2D-Konzept wirken. Für die Bearbeitung der Wirkungsbereiche kann das Aktivieren des In-Place-Modus sehr nützlich sein, mit der unser Charakter auf einen Fleck gezwungen wird. Das bedeutet, dass der Charakter zwar seine zugewiesene Animation abspielt, sich aber selbst bei einer Laufanimation nicht vom Fleck bewegt. **Abbildung 17** zeigt den aktivierten In-Place-Button.

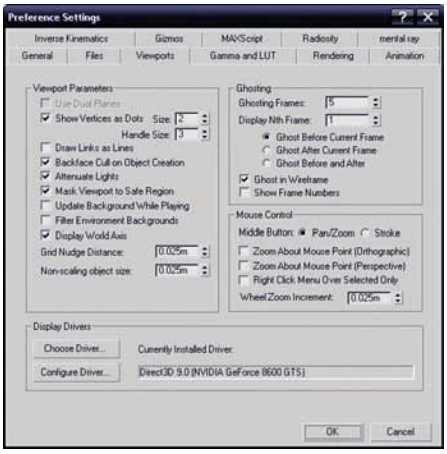


Abbildung 15: Ghosting-Eigenschaften im Register »Viewports«.

Abbildung 17: Aktivieren des In-Place-Modus.

Editieren der Envelopes

Abbildung 18 zeigt den nächsten und wichtigsten Schritt für die Animation unseres Charakters: die Anpassung der Envelopes und deren Einfluss auf das Charakter-Mesh. Neben dem aktiven Button »Edit Envelopes« sehen Sie in der Liste des Skin-Modifikators den ausgewählten Bereich der Hand unseres Bipeds. **Abbildung 19** zeigt uns diesen Bereich am Modell. Hier sehen Sie den Hauptwirkungsbereich der Hand in Rot markiert. Das eigentliche Biped wurde aus Gründen der Übersichtlichkeit ausgeblendet, ist aber noch in der Szene vorhanden. Die Farben markieren die von der ausgewählten Hand unterschiedlich stark beeinflussten Regionen des Arms. Die am stärksten beeinflussten Bereiche sind hierbei ebenfalls rot markiert, wobei die

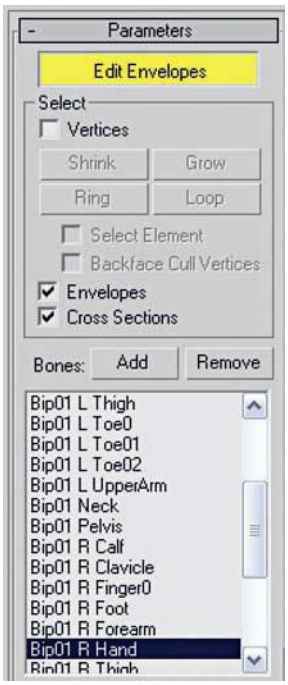


Abbildung 18: Auswahl vom Hand Envelope.

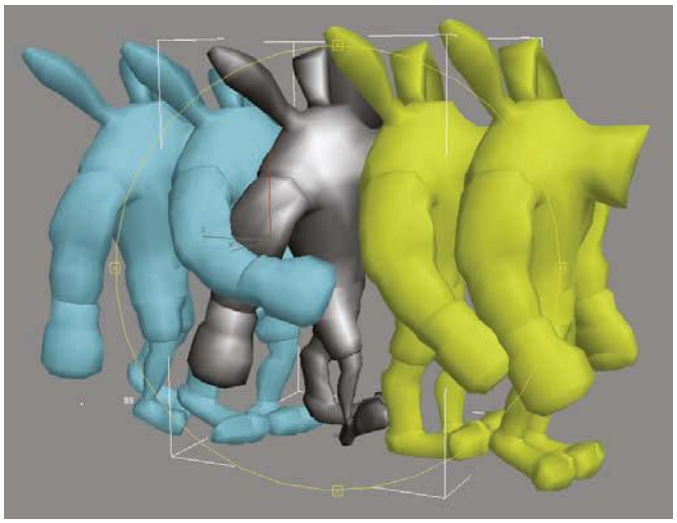


Abbildung 16: Ghosting einer BadAss-Rabbit-Animation.

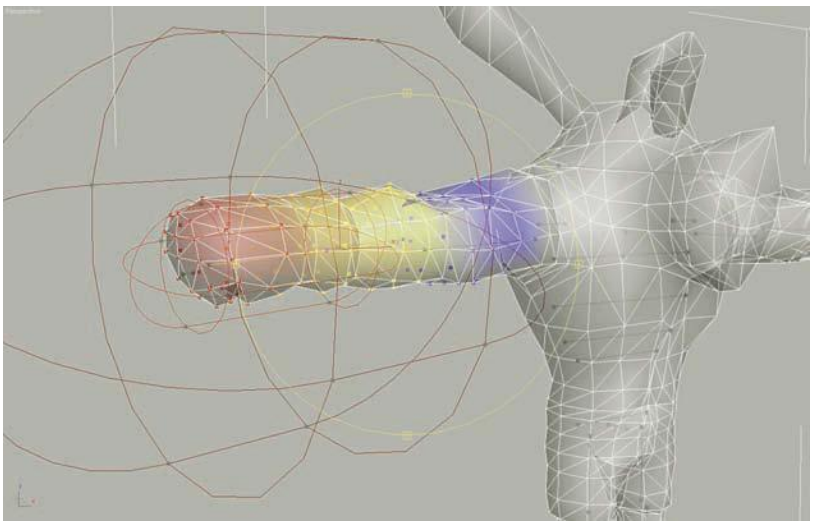


Abbildung 19: Hand Envelope und Wirkungsbereich.

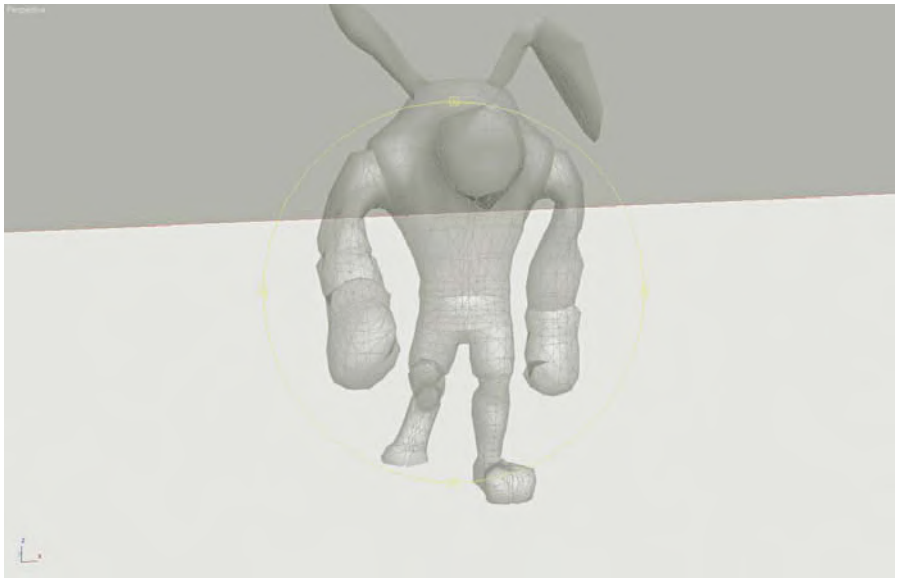


Abbildung 20: Animationsframe vom unbearbeiteten BadAss Rabbit.

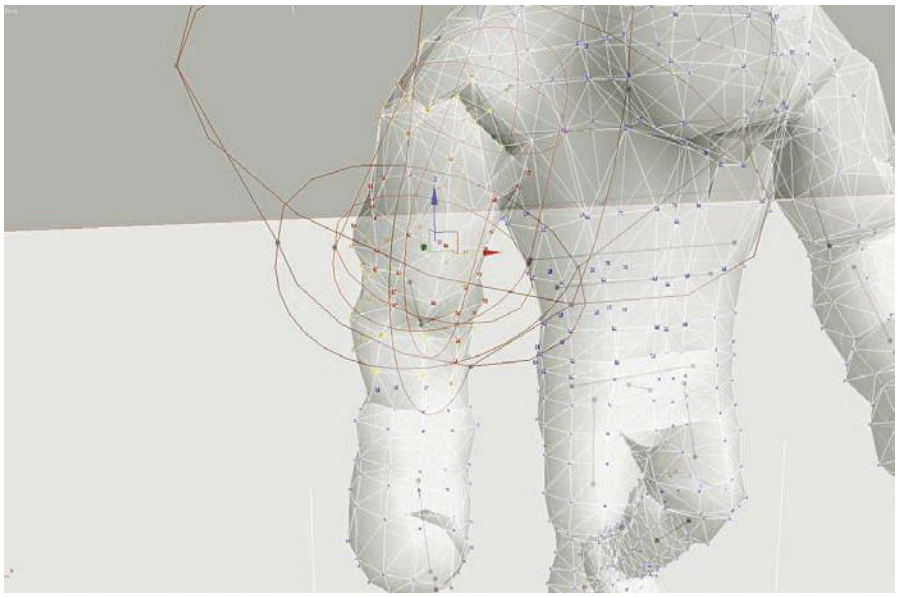


Abbildung 21: Editierte Kräfteeinwirkung des Bipeds am Bizeps.

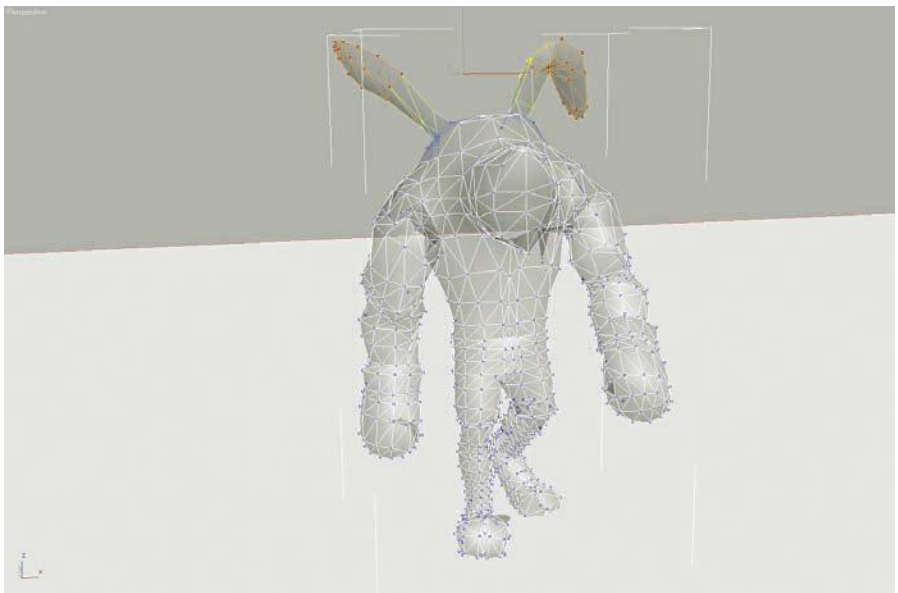


Abbildung 22: Weiche Auswahl der Ohren.

Farbe dann je nach Stärke der Beeinflussung abnimmt. Blau schließlich signalisiert eine sehr geringe Beeinflussung.

In **Abbildung 20** sehen Sie den aktuellen Zwischenstand unserer bisherigen Schritte. Aus dieser Perspektive erkennen Sie eine Vielzahl an verzerrten Körperregionen, wie etwa den Oberarm. Diese Verzerrungen sind auf falsche Einstellungen der Wirkungsbereiche bei den unterschiedlichen Envelopes zurückzuführen. Werden bestimmte Mesh-Bereiche durch zu starke Wirkungsbereiche beeinflusst, können an diesen Stellen Verformungen des Meshs auftreten. Falls das Biped nur sehr ungenau oder gar nicht an das Mesh angepasst wurde, kann es auch vorkommen, dass Punkte des Meshs nicht im Wirkungsbereich des Bipeds sind und somit bei einer Animation unberücksichtigt und im Raum stehen bleiben. In beiden Fällen muss die Stärke der Beeinflussung auf bestimmte Punkte des Meshs überarbeitet werden. Hierzu können Sie die roten Ringe der Envelopes verschieben oder skalieren. Der Skin-Modifikator bietet auch die Möglichkeit, Punkte direkt anzuwählen und ihnen einen bestimmten Wert der Beeinflussung durch den aktuell angewählten Biped-Bereich zuzuweisen. Im Fall des zusammengequetschten Meshs im Bereich der Oberarme aus **Abbildung 20** wählen Sie den Oberarm des Bipeds aus und dann im Skin-Modifikator »Edit Envelope«. Sie werden feststellen, dass die Beeinflussung des Oberarm-Meshs durch den des Bipeds zu schwach eingestellt ist. Wenn Sie die Beeinflussung der Punkte am Oberarm auf nahezu 100 Prozent verstärken, werden Sie sehen, wie sich der Oberarm des Meshs im Viewport ausbeult und wieder seine ursprüngliche Form annimmt. Die Punkte, die stark vom ausgewählten Biped-Bereich beeinflusst werden, sind rot markiert. Je nach Beeinflussung nimmt die Farbmarkierung über Orange und Gelb hin zu Blau ab. Sie sehen den Oberarm und die farblich markierten Punkte des Meshs inklusive Envelope in **Abbildung 21**. Die plötzliche Verformung des Meshs zurück in den Ausgangszustand liegt daran, dass wir durch die Verstärkung der Beeinflussung gleichzeitig andere Bereiche des Bipeds deutlich abgeschwächt haben. Überarbeiten Sie nun alle übrigen Bereiche des Charakters, wie am Oberarmbeispiel gezeigt wird.

Der Flex-Modifikator

In einem weiteren Schritt fügen Sie dem ausgewählten Mesh den Modifikator »Flex« zu, damit Sie die Bewegung der langen Hasenohren möglichst realistisch automatisieren können. Flex verwandelt ausgewählte Mesh-Bereiche in eine Art Wackelpudding, macht diese also flexibel und träge, damit sich das Mesh natürlicher bewegt. Wichtig ist dies für die Ohren unseres BadAss Rabbits, da die langen Löffel von einer Laufanimation beeinflusst werden und mitwackeln sollen. Wählen Sie hierzu das Mesh aus, fügen Sie dann den Modifikator »EditMesh« hinzu. Aktivieren Sie die Punktauswahl und die

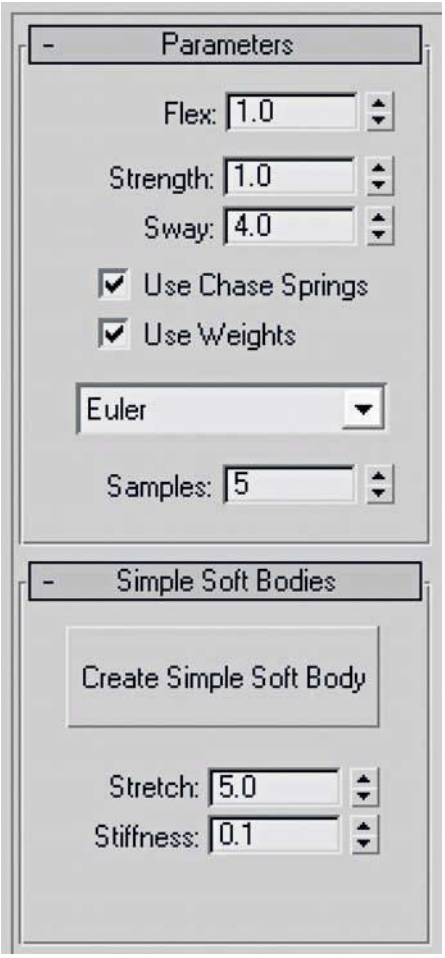


Abbildung 23: Parameter des Modifikators »Flex«.

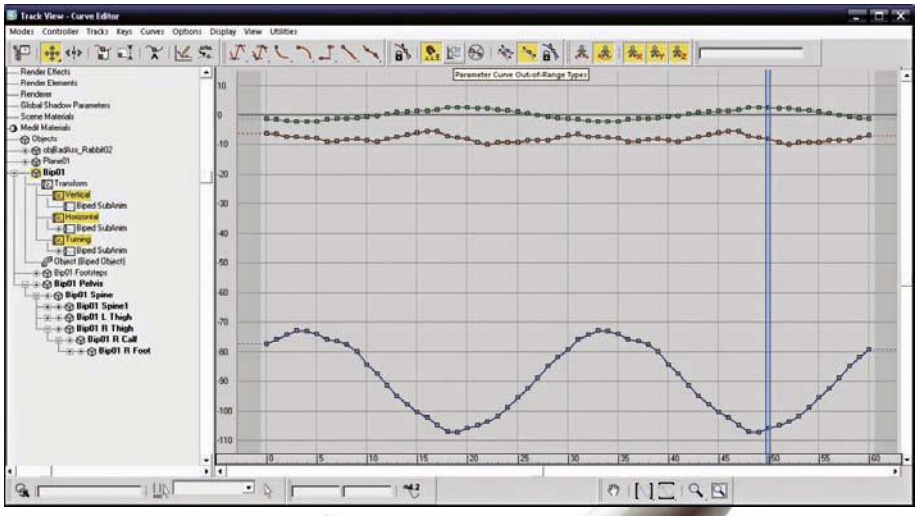
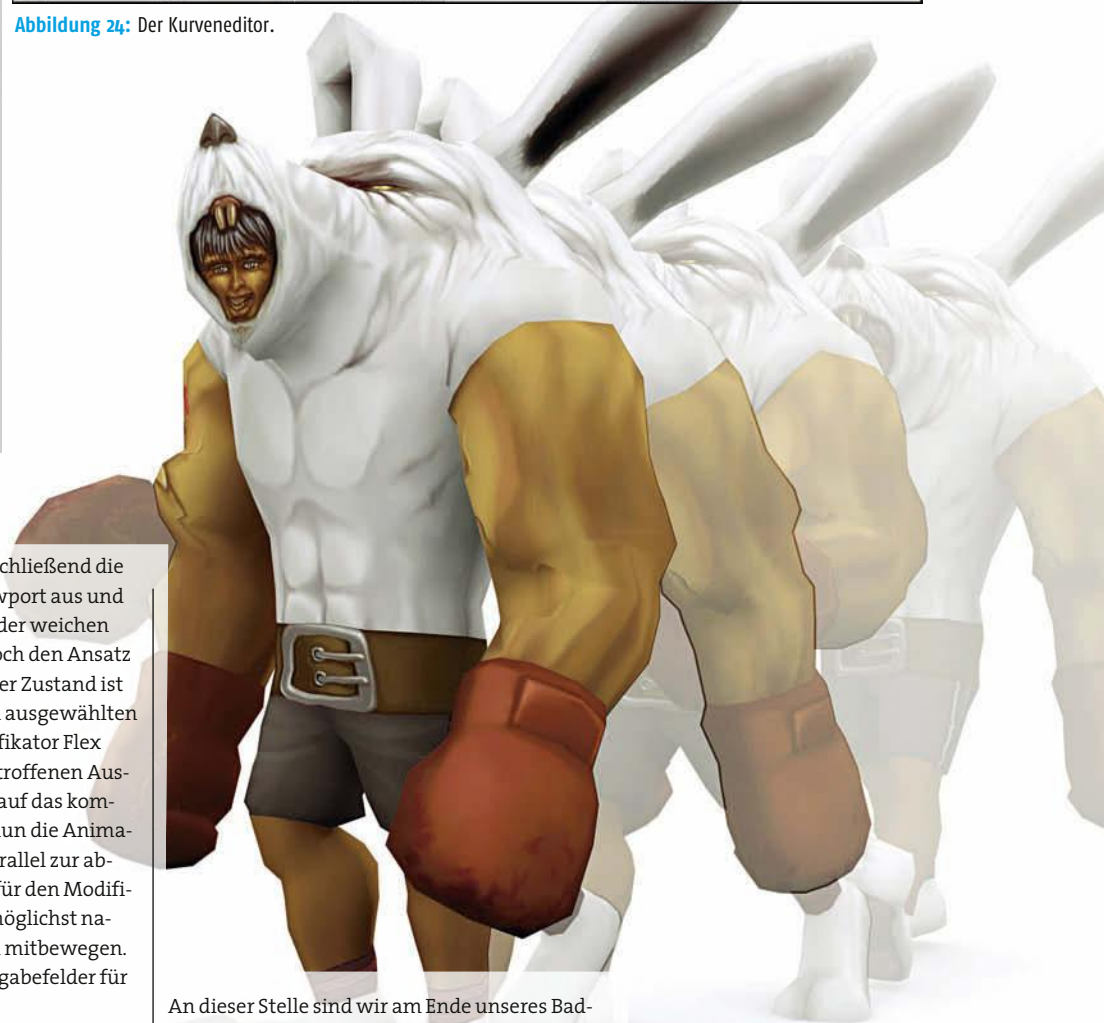


Abbildung 24: Der Kurveneditor.



weiche Auswahl. Wählen Sie anschließend die oberen Punkte der Ohren im Viewport aus und stellen Sie den Wirkungsbereich der weichen Auswahl so ein, dass er gerade noch den Ansatz der Ohren am Kopf umfasst. Dieser Zustand ist in Abbildung 22 zu sehen. Mit den ausgewählten Punkten fügen Sie nun den Modifikator Flex hinzu, der sich dank der zuvor getroffenen Auswahl auf diese Punkte und nicht auf das komplette Mesh bezieht. Starten Sie nun die Animation des Hasen und passen Sie parallel zur ablaufenden Animation die Werte für den Modifikator so an, dass sich die Ohren möglichst natürlich mit den Laufbewegungen mitbewegen. Den Modifikator Flex und die Eingabefelder für seine Werte zeigt Abbildung 23.

Schlussbemerkungen

Einige der Vorteile bei der Arbeit mit Bipeds sind das einfache Anlegen und Bearbeiten von Animationen, die für jedes Biped eingesetzt werden können. Da sich die Formen und Gliedmaßen von Figuren aber meist unterscheiden, ist oft ein Feintuning für geladene Animationen notwendig. Dieses können Sie relativ bequem über die Anpassung mittels des Kurveneditors in 3D Studio Max durchführen. Den Kurveneditor (Abbildung 24) finden Sie in der Hauptmenüleiste des Programms.

An dieser Stelle sind wir am Ende unseres Bad-Ass-Rabbit-Workshops angelangt. Mit den gezeigten Vorgehensweisen sind Sie nun in der Lage, Ihre eigenen 3D-Charaktere zu entwickeln und diese zu animieren. Bitte beachten Sie, dass der Workshop nur einen von vielen Wegen der Charakterentwicklung aufzeigt. Es gibt beispielsweise sowohl andere Modelliermöglichkeiten als auch fortgeschrittene Texturierung, wie beispielsweise die Erstellung von Normal Maps. Aufgrund solcher Maps und der zunehmenden Rechnerleistung wird das Modellieren von High-Poly-Charakteren immer wichtiger.

Chris Noeth

Kleines Gerät, große Hürden
Effiziente DS-Entwicklung

Nintendos Handheld ist eine Plattform der Kompromisse. Unsere Case Study von Giana Sisters DS zeigt, auf welche Kompromisse Sie sich einstellen müssen und welche Sie umgehen können.

Daniel Matzke

ist Gründer und Geschäftsführer von Bitfield.

Daniel arbeitet seit 1999 in der professionellen Spieleentwicklung. Nach dem Abschluss seines Studiums der Informatik im Jahr 2005 gründete er zusammen mit Kollegen die Bitfield GmbH, wo er heute vor allem für die Kernkomponenten der BitEngineDS verantwortlich zeichnet. Parallel ist Daniel an der Games Academy als Dozent und Fachbereichsleiter tätig und engagiert sich als Meeting-Coordinator beim IGDA-Chapter-Berlin.
Daniel.Matzke@bitfield.de

David Salz

ist Gründer und Geschäftsführer von Bitfield.

David ist Diplom-Informatiker und arbeitet seit neun Jahren in der Spieleentwicklung. Dabei wirkte er unter anderem an Titeln wie Wiggles, Paraworld und Singles mit. Er ist Dozent und Fachbereichsleiter an der Games Academy in Berlin sowie einer der Organisatoren des IGDA-Chapters Berlin. Vor dreieinhalb Jahren machte er sich mit der Bitfield GmbH selbstständig und entwickelt heute am liebsten Tools und Editoren.
David.Salz@bitfield.de

WWW
makinggames.de

- Handheld-Alternative: Software Development für das iPhone

Abbildung 1: Giana Sisters 1987 auf dem C64 (links) und 2009 auf dem Nintendo DS (rechts).

Das Spiel »The Great Giana Sisters« dürfte vielen Lesern noch aus ihrer Kindheit bekannt sein. 1987 als Super-Mario-Klon für den C64 entwickelt, erlangte das Jump & Run schnell Kultstatus. Heute, mehr als 20 Jahre nach dem Original, feiert das Spiel ein Comeback auf dem Nintendo DS (siehe **Abbildung 1**). Am Beispiel dieser Neuauflage möchten wir zeigen, welche Hürden bei der Entwicklung von DS-Titeln auftauchen und wie Sie diese am effizientesten überwinden.

Tile-Systeme

Wer schon einmal mit einem Nintendo DS oder dessen Vorläufern gearbeitet hat, dürfte sich auf kurz oder lang auch mit dem Prinzip einer Tile Engine beschäftigt haben.

Die grundlegende Idee dieser Technologie besteht darin, große Grafiken aus kleinen quadratischen Kacheln zusammenzusetzen, anstatt sie als komplette Bitmap zu speichern. So werden redundante Daten in einer Grafik zusammengelegt. Das spart in erster Linie Grafikspeicher, ermöglicht jedoch auch einige Tricks, die mit Bitmap-basierten Grafiken nicht möglich sind.

Auch Giana Sisters DS verwendet Tile-Grafiken. Die Parallax-Ebenen im Hintergrund (Berge, Wolken ...) wurden durch einen 3D-Grafik-Layer realisiert. Im Unterschied dazu besteht die Ebene, in der das Spiel abläuft, aus drei Tile Layern, die synchron scrollen und so optisch eine Einheit bilden. Diese drei Ebenen – Foreground Layer, Gameplay Layer und Background Layer – ermöglichen es, die Spielfigur vor bzw. hinter bestimmten Elementen des Levels einzufügen. Während die vordere und die hintere Ebene nur zur Dekoration des Levels dienen, wird der Gameplay Layer zusätzlich auch für die Spielsimulation verwendet (siehe **Abbildung 2**).

Für jede Grafik-Kachel auf der Simulations-Ebene wird gleichzeitig auch eine analoge Kachel mit Kollisionsinformationen angelegt. Die eigentlichen Kollisionsdaten bestehen aus einer Flagkombination pro Pixel, sodass sich unterschiedliche Hindernisse wie Wände, schiefe Ebenen oder Wasser unterscheiden lassen (siehe **Abbildung 3**).

Tile-Vorteile

Durch die Nutzung des Tile-Prinzips für die Kollisionsdaten ergeben sich mehrere Vorteile: Wie bei den Grafikkacheln spart man Speicher, da redundante Levelelemente zusammengelegt werden. Da es neben den Grafik- und Kollisionskacheln letztlich nur eine gemeinsame Level-Definition gibt, hängt der zusätzliche Speicherbedarf für die Kollision nicht von der Größe des Levels ab. Gemeinsam genutzte Datenstrukturen sparen aber nicht nur Speicher: Im Fall von Giana Sisters DS hat es auch den Effekt, dass der Level-Designer bei der Erstellung des Levels gleichzeitig die Kollisionsinformation erstellt hat. Rein grafische Elemente, die wiederum keinen Einfluss auf das Spielgeschehen haben sollten, wurden dagegen im Foreground bzw. Background Layer platziert. Da das Gamedesign von Giana Sisters vorsieht, dass auch der Level selbst beim Spielen in gewissem Maße verändert werden kann, hat die Zusammenlegung von Kolli-

sions- und Grafikdaten den Effekt, dass eine Änderung der Levelgrafik in einem Bereich auch die Kollision in diesem Bereich ändert.

Der zweite positive Effekt von Tile-Systemen ist, dass die Manipulation der Kachel-Definitionen selbst Auswirkungen auf alle Bereiche des Levels hat, die diese Kacheln referenzieren. So lassen sich zur Laufzeit durch Manipulation weniger Datenmengen auch großflächige Bereiche eines Levels animieren. Dieser Effekt lässt sich beispielsweise gut für Wasser- oder Lavaflächen verwenden. In Giana Sisters DS wurden auf diese Weise auch Spiel-Elemente wie Diamanten oder Item-Boxen animiert, wodurch die Levels allgemein viel lebendiger wirken. Da zu jedem Zeitpunkt immer nur eine Animationsphase sichtbar ist, wird durch diesen Effekt auch kein zusätzlicher Videospeicher verbraucht.

Für die Erstellung von Tile-basierten Grafiken sind traditionelle Grafik-Tools wie Photoshop eher ungeeignet. Für Giana Sisters DS wurden daher angepasste Editoren entwickelt, die das Potenzial dieser Technik voll ausschöpfen können.

Tools

Tools sind Multiplikatoren. Ein gutes Tool kann also sehr viel Zeit sparen und die Arbeitsleistung der Anwender erhöhen. Ein schlechtes Tool hingegen kostet Zeit und Nerven. Gute Tools ermöglichen dem Team, neue Ideen sofort auszuprobieren und Änderungen schneller zu testen.

Wo immer möglich sollte auf vorhandene Standardsoftware zurückgegriffen werden, z.B. Photoshop, 3D Studio Max oder Maya. Auch Konsolenhersteller bieten spezialisierte Tools für ihre jeweilige Plattform an. Es gibt jedoch in fast jedem Projekt Bereiche, in denen Standardsoftware nicht mehr ausreicht und eigene Lösungen entwickelt werden müssen. Giana Sisters DS dürfte ein relativ typischer Fall sein: Level- und Karteneditor waren komplette Eigenentwicklungen (**Abbildung 4**) und verschlangen etwa 30 Prozent des Programmierbudgets. Eine solche Entscheidung muss gut überlegt sein, denn neben den erheblichen Kosten ist zu bedenken, dass die eigenen Tools dem Team nicht von Anfang an zur Verfügung stehen, sondern zunächst einige Wochen oder Monate Entwicklungszeit benötigen. Es ist zu überlegen, ob und wie diese Zeit mit Standard-Tools sinnvoll überbrückt werden kann.

Tool-Kompatibilität

Wenn ein eigenes Tool unumgänglich ist, sollte auf möglichst große Kompatibilität zu existierenden Programmen und Arbeitsabläufen geachtet werden. GUI und Bedienung sollten der Standardsoftware entsprechen, aber es ist beispielsweise auch leicht möglich, existierende Datenformate zu unterstützen. Die Dateiformate der meisten Tools – selbst bei proprietären Tools der Konsolenhersteller – sind dokumentiert und lassen sich mit eigenen Programmen lesen und schreiben. Manche Standardformate sind sogar auf Erweiterbarkeit ausgelegt, sprich um eigene

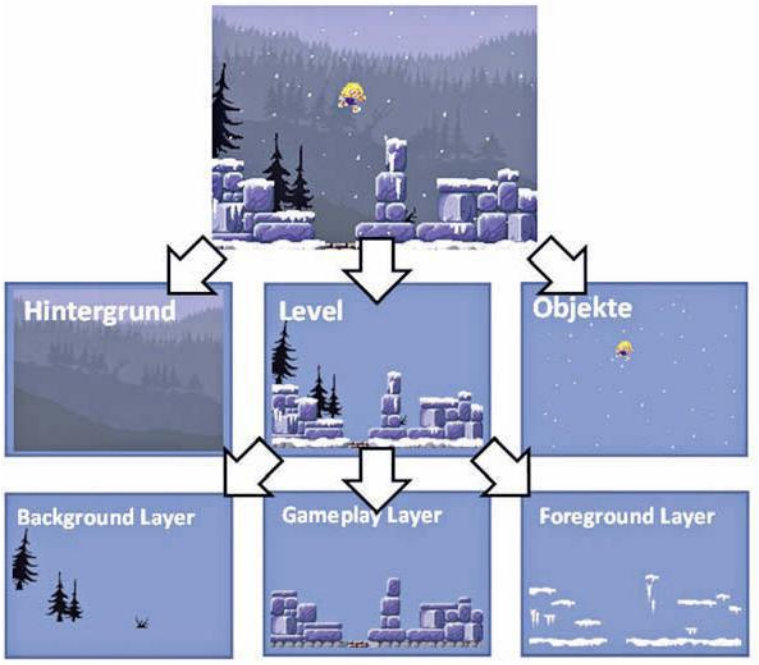


Abbildung 2: Die Spielgrafik setzt sich aus insgesamt fünf Ebenen zusammen.

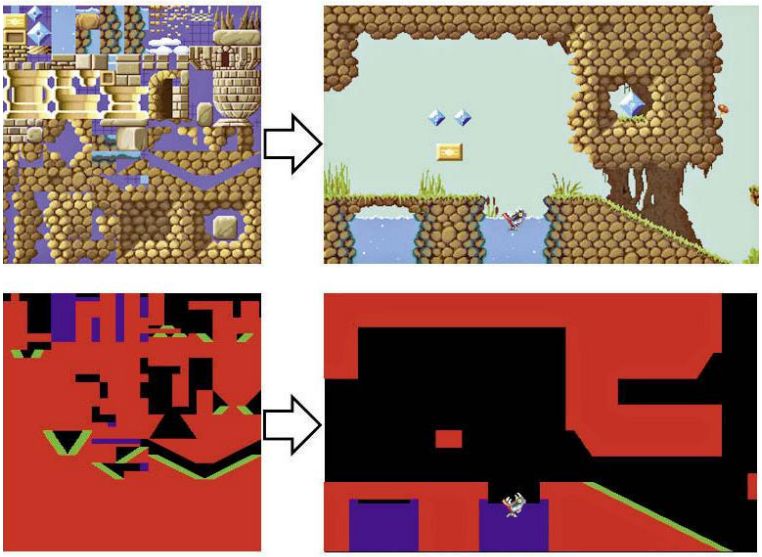


Abbildung 3: Alle Levels sind aus kleinen Kacheln (Tiles) gebaut. Zu jeder Grafik-Kachel existiert auch eine weitere Kachel, die Kollisionsinformationen enthält.

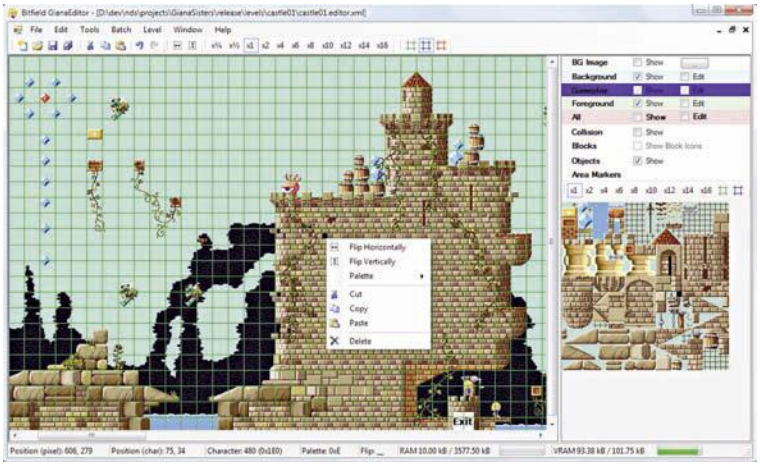


Abbildung 4: Standardsoftware hat viele Vorteile. Eigenentwicklungen, wie etwa der Giana-Sisters-Leveleditor, sind in vielen Fällen jedoch die bessere Lösung.

Abbildung 5: Dank der Kompatiblität erzeugt der Leveleditor auf Wunsch Panorama-Screenshots.

Datensätze erweiterbar. Wenn eine Standardsoftware eine bestimmte Funktionalität nicht bietet, ist das also kein Grund, ein völlig neues Tool mit allen Funktionen und eigenem Dateiformat zu entwickeln. Oftmals tut es auch ein eigenes Plug-in oder ein kleines Zusatzprogramm, das die vorhandenen Dateien lesen, bearbeiten und wieder speichern kann. Dies hat zudem den Vorteil, dass der existierende Code auf dem Endgerät weiterverwendet werden kann.

Der Screenshot-Trick

Der Giana Sisters DS Leveleditor verwendet die gleichen Dateiformate wie unsere DS-Engine und ist zu existierenden Standard-Tools kompatibel. Die Levels werden im Editor auf dem PC exakt so wie auf dem DS dargestellt. Wir konnten den Leveleditor daher verwenden, um Panorama-Screenshots kompletter Levels anzufertigen; auf dem DS wäre das so nicht möglich gewesen. Diese Funktion war für uns insbesondere beim Testing eine große Hilfe. Für jeden Level lieferten wir der QA-Abteilung drei Panorama-Screenshots (siehe **Abbildung 5**): Erstens den kompletten Level, wie der Spieler ihn im fertigen Spiel sieht; eine zweite Version mit farbig hervorgehobener Kollisionsinformation, und drittens eine Variante, in der Geheimgänge und versteckte Bonus-Items markiert waren (siehe **Abbildung 6**). Der Leveleditor konnte das Screenshot-Paket für alle Levels auf Knopfdruck automatisch erstellen, sodass wir ohne Aufwand zu jeder neuen Testversion aktualisierte Bilder liefern konnten. Für die Tester war dies eine große Hilfe, da sie nicht umständlich nach Geheimgängen und Items suchen mussten und gegenüber der letzen Version veränderte Levelabschnitte auf den Bildern schnell zu erkennen waren.
Gefundene Fehler oder Verbesserungsvorschläge wurden direkt in den Screenshots markiert und an die Level Designer zurückgeschickt. Grafik- oder Kollisionsfehler waren auf den Bildern oft sogar um einiges leichter zu finden als im Spiel selbst. Letztendlich verwendeten wir die Panorama-Screenshots sogar für Pressematerial und Marketingzwecke.

Ressourcen überwachen

Eine weitere sehr bewährte Funktion unseres Editors war die Ressourcenanzeige. Jedes im Level platzierte Monster oder Item verbraucht eine bestimmte Menge Hauptspeicher (RAM) und Videospeicher (VRAM). Die auf dem DS vorhandenen Ressourcen sind jedoch knapp bemessen und das Limit ist schnell erreicht. Der Editor konnte daher per Balkendiagramm die RAM/-VRAM-Belegung eines Levels anzeigen und den Level Designer warnen, sobald eine Grenze überschritten wurde. Die Informationen zum Speicherverbrauch einzelner Monster und Objekte mussten von den Programmierern manuell gepflegt werden, da der Speicherverbrauch teilweise im Code begründet lag und sich nicht allein aus der Größe von Grafikdateien berechnen lässt. Die Speicherverbrauchsanzeige im Leveleditor konnte folglich nicht immer hundertprozentig genau sein, war aber nichtsdestotrotz eine wertvolle Hilfe.

Abbildung 6: Der Editor erzeugt verschiedene Panorama-Screenshots für die Tester: normale Ansicht (links), mit Kollisionsinformationen (Mitte) und mit markierten Secrets (rechts).

Sounds und Musik

Sounddaten belegen von Natur aus sehr viel Speicherplatz, sowohl auf dem Datenträger als auch im RAM. Bei Giana Sisters DS werden fast 50% des Moduls und zeitweise bis zu 30% des Arbeitsspeichers von Musik und Soundeffekten beansprucht. Der hohe Anteil am RAM hat zwei Gründe:

Erstens können Soundeffekte in der Regel nicht dynamisch nachgeladen werden, da sie bei Bedarf sofort abgespielt werden müssen und das menschliche Gehör selbst die kleinste Verzögerung sofort registrieren würde. Alle für einen Level benötigten Sounds müssen deshalb bereits beim Start des Levels geladen sein. Giana Sisters DS definiert für verschiedene Monster, Levelelemente und Sonderwaffen eigene Soundbänke, d.h. Sammlungen von Soundsamples. Geladen werden dabei jeweils nur die Bänke, die in einem Level tatsächlich vorkommen.

Der zweite Grund für den hohen Arbeitsspeicherbedarf ist die Musik. Ein Streaming-Format wie z.B. MP3 ist für kleine Datenträger oder Download-Games noch immer viel zu groß. Formate wie MIDI, MOD oder XM sind erheblich kleiner, müssen aber komplett in den Arbeitsspeicher geladen werden. Ob man sich nun für MIDI oder MOD/XM entscheidet, bleibt Geschmackssache. Tatsächlich sind sich diese Formate technisch sehr ähnlich, denn alle verwenden Noten und gesampelte Instrumente, die auf verschiedene Tonhöhen gepitcht werden können. Entsprechend einfach ist es, zwischen diesen Formaten hin- und herzukonvertieren. Die Musik für Giana Sisters DS wurde auf dem PC als XM komponiert und dann mit dem (sehr preisgünstigen) Tool 2Midi in MIDI konvertiert. Wichtiger als das Dateiformat ist eine effiziente Speichernutzung. Samples sollten generell komprimiert werden, etwa im ADPCM-Format. Instrumenten-Samples werden in der Regel von mehr als einem Musikstück verwendet, daher sollten alle Musikstücke auf einen gemeinsamen Pool von Samples zurückgreifen. Diese Optimierungen werden nicht von allen Playern unterstützt, sollten aber ein wesentliches Auswahlkriterium sein. Giana Sisters DS bringt so 23 Musikstücke in gerade einmal 2,5 MB unter.

Für den Komponisten ist es wichtig, das Finetuning der Musik direkt auf der Zielhardware abstimmen zu können. Daher haben wir ein Tool gebaut, mit dem unser Musiker seine Daten mit einem Klick selbst konvertieren und auf dem DS anhören konnte. Das Tool zeigt außerdem den Speicherbedarf der Musikstücke an, sodass schon beim Komponieren auf die Einhaltung des Budgets geachtet werden kann.

Datenkompression

Nicht nur der Arbeitsspeicher ist eine begrenzte Ressource, sondern auch der Platz auf dem Datenträger – das gilt umso mehr für Downloadspiele. Entsprechend naheliegend ist die Idee, Daten komprimiert zu speichern und beim Laden zu entpacken. Die auch für uns etwas überraschende Erkenntnis war, dass man durch die Wahl eines passenden Kompressionsverfahrens sowohl Speicherplatz sparen als auch Ladezeiten reduzieren kann. Die komprimierten Daten sparen also mehr Ladezeit ein, als für das Entpacken benötigt wird. Dank Kompression muss man auch vor praktischen, aber »großen« Dateiformaten wie XML keine Angst mehr haben. Giana Sisters DS bringt knapp 27 MB unkomprimierte Daten auf weniger als 8 MB unter und passt so auf einen deutlich kleineren Datenträger. Ein nicht zu unterschätzender Kostenfaktor!

Problem Speicherfragmentierung

Zum Schluss möchten wir uns noch einem sehr technischen Thema zuwenden: der Speicherfragmentierung. Arbeits- und Videospeicher kann – genau wie eine Festplatte – durch ständiges Erzeugen und Löschen von Objekten fragmentieren (siehe **Abbildung 7**). Auf dem PC fällt dieses Problem dank des riesigen Arbeitsspeichers und Disc Swapping meist nicht ins Ge-

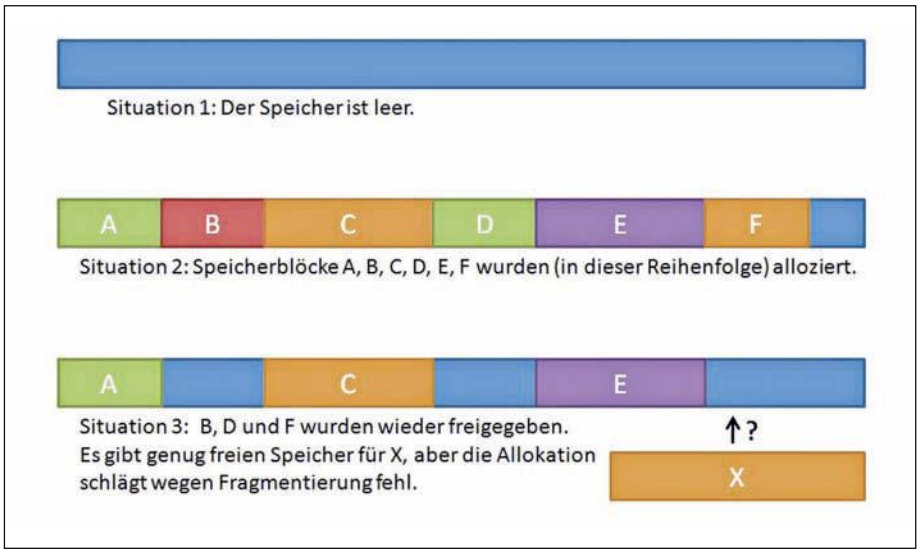


Abbildung 7: Gerade bei geringem verfügbaren Speicher ist Fragmentierung häufig ein Problem.

Bitfield

Bitfield ist ein Entwicklungsstudio aus Berlin. Neben Auftragsarbeiten und Eigenproduktionen für PC und DS ist Bitfield auch offizieller »Nintendo 3rd Party Tool Developer«.

Die BitEngineDS ist Bitfields Allzweck-Game-Engine für Nintendo DS und DSi und kommt bereits in mehr als 40 Produktionen weltweit zum Einsatz.

Zum Lieferumfang der Engine gehören neben ausführlichen Beispielen und Tutorials auch erweiterte Versionen der Tools, die für Giana Sisters DS eingesetzt wurden – insbesondere sind das ein 2D-Leveleditor und ein komplettes Lokalisationssystem. Bitfield bietet auch Coaching und Consulting zum Thema DS-Entwicklung an. Die BitEngineDS kann auf Wunsch kostenlos evaluiert werden.

www.bitfield.de / engine@bitfield.de

wicht; auf einer kleinen Plattform hingegen müssen unbedingt Gegenmaßnahmen ergriffen werden.

Die ersten Allokationen belegen den Speicher meist sequenziell von vorn nach hinten. Belegter und freier Speicher bilden jeweils einen kompakten Block – das ist die Idealsituation. Sobald nun aber einige Speicherblöcke wieder freigegeben werden, während andere weiterhin belegt bleiben, entstehen Lücken, d.h., der freie Speicher ist nicht mehr zusammenhängend. Nun kann es zu einer Situation kommen, in der für eine neue Allokation zwar in der Gesamtsumme genug freier Speicher zur Verfügung steht, dieser freie Speicher aber fragmentiert, also nicht zusammenhängend ist. Die Allokation schlägt daher fehl. Die meisten Speichermanager versuchen Fragmentierung zu vermeiden, indem sie für neue Allokationen den von der Größe her am besten passenden freien Speicherblock verwenden. Da es aber meist keinen genau passenden Speicherblock gibt, nimmt die Fragmentierung über die Zeit in der Regel zu, d.h., es werden immer mehr immer kleinere freie Speicherblöcke. In diesem Zusammenhang muss auch auf Speicherlecks (Memory Leaks) hingewiesen werden, also allozierter Speicher, der schlicht »vergessen« und nicht wieder freigegeben wird. Leaks fressen den auf dem DS ohnehin knappen Speicher mit der Zeit immer mehr auf und erhöhen gleichzeitig die Fragmentierung. Meist wird die Fragmentierung zum Problem, noch bevor der Speicher wirklich voll ist.

Diese Art von Problemen ist deshalb so unangenehm, weil die Fehler schwer reproduzierbar sind und meist erst nach einer gewissen Spielzeit auftreten. Während der Entwicklung von Giana Sisters DS hatten wir beispielsweise einen Leak- und fragmentierungsbedingten Absturz bei einem Bosskampf, der aber nur auftrat, wenn vorher ein bestimmter Level gestartet, jedoch nicht zu Ende gespielt wurde.

Lösungsansätze

Wie erkennt und vermeidet man Leak- und Fragmentierungsprobleme? Zunächst sollte man neben der Kennzahl »freier Speicher« auch die Zahl »größter freier Block« im Auge behalten (gute Speichermanager bieten diese Funktion an). Weichen diese beiden Zahlen stark voneinander ab, hat man ein Fragmentierungsproblem. Am besten gibt man sie in regelmäßigen Abständen aus – zum Beispiel immer bei Levelstart und Levelende – und beobachtet die Entwicklung über mehrere Stunden Spielzeit.

Speicherlecks findet man, indem der Speichermanager alle Allokationen protokolliert und an fest definierten Punkten überprüft, ob alle ordnungsgemäß wieder freigegeben worden sind. Bei PC-Anwendungen findet dieser Check meist nach dem Beenden statt. Konsolenspiele müssen sich aber meist gar nicht beenden können, sondern laufen weiter, bis buchstäblich der Strom abgestellt wird. Zu Debugging-Zwecken lohnt es sich aber, ein Herunterfahren mit vollständiger Speicherfreigabe einzubauen; erfahrungsgemäß macht das viele versteckte Probleme sichtbar. Ein weiterer guter Trick ist, alle Ressourcen und Allokationen in Gruppen zu ordnen. Giana Sisters DS unterscheidet unter anderem die Gruppen »User Interface« und »Level«. Alle Speicherallokationen, die zu einem Level gehören, werden als »Level« markiert und müssen beim Verlassen des Levels wieder freigegeben worden sein. Daten in der Gruppe »User Interface« hingegen dürfen bis zum Beenden des Spiels im Speicher bleiben.

Ein weiteres effektives Mittel gegen Fragmentierung ist, verschiedene Speichermanager für verschiedene Zwecke zu verwenden. Fragmentierung entsteht, weil langlebige und kurzlebige Objekte aus dem gleichen Pool alloziert werden. Kurzlebige Objekte reißen Lücken, langlebige zementieren die Lücken. Insofern ist es logisch, getrennte Speicherbereiche für diese Gruppen zu verwenden. Auch inhaltliche Trennungen können sinnvoll sein. Giana Sisters DS verwendet beispielsweise getrennte Speichermanager für Sounddaten, User-Interface-Daten, Leveldaten und Gianas Animationen. Auf diese Weise kann etwa der Soundspeicher nicht von Leveldaten fragmentiert werden. Der Speicher für Leveldaten wird immer beim Levelstart gefüllt und beim Levelende komplett geleert, kann also in sich nicht fragmentieren. Gianas Animationsdaten sind so umfangreich, dass sie teilweise im laufenden Level nachgeladen werden müssen, was aber keinen der anderen Speicherbereiche beeinflussen kann.

Daniel Matzke, David Salz

PNY™

MIT DEM RICHTIGEN WERKZEUG **WIRD DARAUS EIN WELTSTAR.**

NVIDIA QUADRO

Mit uns werden Kinokassen gestürmt, Maschinen konstruiert, Wissenschaft beflügelt, Unternehmen verwaltet, Finanzströme bewegt, Häuser gebaut, Fernsehen möglich und Gamer glücklich gemacht – alles in Echtzeit und mit überragender Performance.

GRAFIK- UND HIGH-PERFORMANCE-COMPUTING-LÖSUNGEN VON PNY – PROFESSIONAL SOLUTIONS.

PNY Technologies Quadro GmbH | Schumanstraße 18a | 52146 Würselen | Tel.: +49 (0)2405 40848-0 **www.pny.de/quadro**

Das Universum erweitern
Addon-Auslieferung in Eve Online

Die Herausforderung: eine gigantische MMO-Erweiterung über die Datenautobahnen schaufeln, und das mit möglichst geringen Auswirkungen auf den Spielbetrieb. Die Jungs von CCP erklären, wie sie es geschafft haben.

In den **2.500 neuen Sonnensystemen** des Addons Apocrypha warten viele neue Inhalte auf die Spieler.

Kurt van Meter

ist Eve Quality Assurance Manager bei CCP in Reykjavik, Island.

James Wyld

ist Virtual World Administrator bei CCP in Reykjavik, Island.

WWW makinggames.de

- Wie die Spieler die Weiterentwicklung von Eve Online mitgestalten
- Digitale Distribution: Welche technische Infrastruktur Sie benötigen

Das Erste, was mir nach einem ausgedehnten Schlaf zu Ohren kommt, sind aufgeregte Gespräche über große Veränderungen im bekannten Universum. Sofort meldet sich mein Forscherdrang und ich beschließe, den Gerüchten mit Hilfe meines altgedienten Schlachtkreuzers auf den Grund zu gehen. Ich docke ab und suche mit den Sensoren nach einer der angeblich überall aufgetauchten Anomalien. Und tatsächlich bemerke ich fast sofort ein ungewöhnliches Signal. Ich nähere mich vorsichtig und traue meinen Augen kaum: Ein Wurmloch hat sich im System geöffnet. Ob sich das Risiko lohnt, allein in unbekanntes Gebiet vorzudringen? Egal, Forscherdrang und Neugier überwältigen jeden Zweifel und ich manövriere mitten in die Anomalie.

Ein kurzer Moment der Orientierungslosigkeit, und der Raum um mein Schiff gleicht einem Wirbel aus Gasen, blaugrün gefärbt von den umliegenden kosmischen Nebeln. Meine Sonden finden eine unbekannte Signatur, ich mache mich sofort auf den Weg. Eine unvorsichtige Dummheit, wie sich nach einer Minute der Stille herausstellt. Fünf Raumschiffe, wie ich sie noch nie zuvor gesehen habe, jagen auf meinen Kreuzer zu und eröffnen sofort das Feuer.

Nach einigen brenzligen Momenten übernehmen lange trainierte Reflexe die Kontrolle und mein Schlachtkreuzer macht seinem Namen alle Ehre. Kurz darauf warten fünf seltsame Wracks darauf, geplündert zu werden. Eine erste Untersuchung zeigt nie gesehene Materialien und Bauteile. Beim Gedanken an das Vermögen, das ich den Forschern zu Hause dafür abnehmen werde, klopfe ich mir selbst auf die Schulter und muss mich beherrschen, nicht dämlich zu grinsen. Doch was war das? Der Scanner warnt mich vor schnüffelnden Sonden. Nichts Unbekanntes diesmal, sondern Sonden der gleichen Art, wie ich sie benutzt habe, um die unbekannten Schiffe aufzuspüren. Jäh verfliegt meine Selbstbeflissenheit und ich jage die Maschinen meines treuen Kreuzers weit über die zulässigen Spezifikationen hinaus. Zu spät. Die erste Laser-Salve schlägt ein, gefolgt von Raketen. Nur die Rettungskapsel bewahrt mich vor einem schnellen Tod. Eben noch Millionär, nun mittellos. Doch ich verdränge meine aufkommende Verzweiflung, selbst in bisher unbekannten und neu entdeckten Gebieten bleibt das Eve-Universum harsch und kalt. Zeit, meine Leute zusammenzutrommeln und es den Schweinen heimzuzahlen.

Das ist ein typisches Beispiel, wie Spieler ihren ersten Tag mit der am 10. März 2009 online gegangenen Erweiterung Apocrypha für Eve Online erlebten. Für die Spieler war das eine ziemlich nahtlose Erfahrung – nach einer etwas verlängerten Downtime fanden sie eine massiv veränderte Spielwelt vor. Hinter den Kulissen ging dem aber ein massiver Aufwand voraus, nicht nur bei der Entwicklung, sondern auch und vor allem beim Ausliefern von Apocrypha.

Flexibilität beim Entwickeln und Testen

Apocrypha ist CCPs erste Erweiterung, die komplett mit der Scrum-Methode entstanden ist. Das bedeutet, dass die Entwickler in kleine Teams aufgeteilt werden, die jeweils für einen spezifischen Bereich zuständig sind. Alle neuen Funktionen und Inhalte werden innerhalb der Teams getestet und nur dann weitergereicht, wenn alle vorher festgelegten Kriterien erfüllt sind. Die Arbeiten werden jeweils in zwei Wochen langen »Sprints« ausgeführt. Dabei legen die Teams vorher fest, was sie in diesen zwei Wochen erledigen. Jeder Sprint baut auf den vorigen Sprint auf und führt am Ende zum fertigen Produkt der jeweiligen Teams.

Die Entwicklung individueller Features in relativer Isolation bereitet häufig Probleme, sobald die Arbeit eines Teams ins Gesamtprogramm integriert werden soll. Dazu kommt die Gefahr, dass neu entwickelte Programmteile nach der Integration Probleme mit anderen Teilen der Software verursachen. Diese Gefahren sind einfach zu übersehen, wenn man die einzelnen Teile nur individuell testet.

Um dieses Risiko zu minimieren, integrierten wir regelmäßig die Arbeiten sämtlicher Teams ins Programm. Während jedes Sprints testeten wir gründlich den kompletten Build. Das beinhaltete auch sogenannte »Regression Tests«. Das sind speziell entwickelte Tests, die so viel Spielfunktionalität wie möglich abdecken, um sicherzustellen, dass wir nicht aus Versehen bereits vorhandene Features durch Neuentwicklungen beschädigt hatten. Organisiert wurden diese Tests zwar intern, ausgeführt hat sie aber ein externer Partner. Durch regelmäßige Plattform- und Performance-Tests vergewisserten wir uns, dass der Client problemlos auf einer Vielzahl von Windows- und Apple-PCs lief. Diese wurden teils intern, teils extern gemacht.

Ein zentraler Punkt bei diesem Prozess war, dass wir in den späteren Entwicklungsphasen jeden fertigen Abschnitt (Sprint) von Apocrypha auf Singularity, unseren Testserver aufspielten. So konnten die Spieler am Test teilhaben und die neuen Features ausprobieren. Zusätzlich attackierten auch unsere offiziellen Bug-Jäger – ein gut organisiertes Team aus fanatischen Eve-Spielern – den jeweils aktuellen Build auf dem Testserver. Bestätigte Fehler wurden dann an unsere Quality Assurance gemeldet und den jeweiligen Scrum-Teams zur Lösung im nächsten Sprint übergeben.

Auslieferung auf DVD

Die Veröffentlichung von Apocrypha war ungewöhnlich für CCP, da wir die Erweiterung mit Hilfe von Atari zeitgleich auch in die Läden bringen wollten. Alle früheren Addons sowie das Spiel selbst gab es bislang ausschließlich als Download. Das resultierte in deutlich mehr Komplexität bei Entwicklung und Produktion, aus mehreren Gründen. Zunächst bedeutete die gleichzeitige Veröffentlichung natürlich, dass wir auf einen Erscheinungstermin festgelegt waren. Während wir zwar immer versuchen, pünktlich zu bleiben, gibt es bei digitaler Auslieferung doch die Möglichkeit einer kurzen Verschiebung, falls im letzten Moment größere kritische Fehler auftauchen sollten.

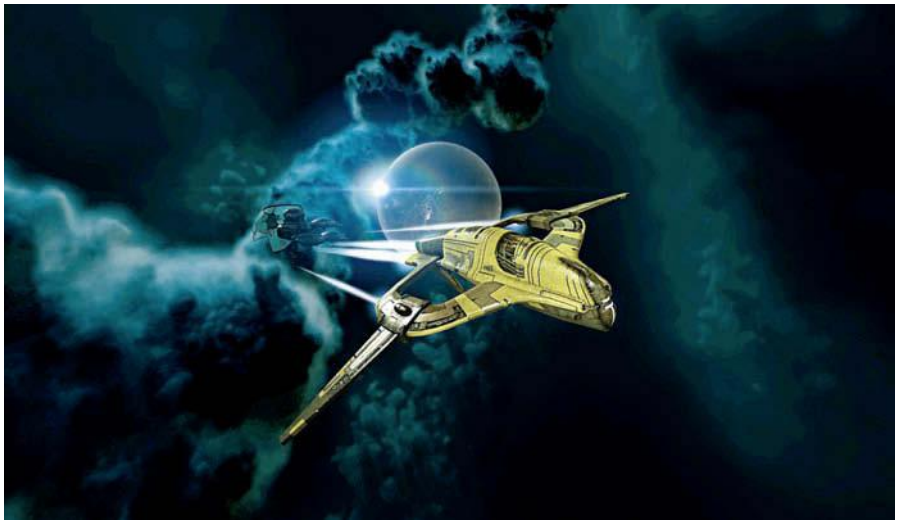
Zusätzlich bedeutete die pure Größe der Erweiterung (besonders die verbesserte Grafik für Apple-Benutzer trug ihren Teil dazu bei), dass wir unseren Installations- und Patch-Mechanismus neu gestalten mussten. Folglich waren die Tests für die DVD-Version und den Patch-Vorgang gewaltig. Wir mussten sicherstellen, dass wir alle älteren Spielversionen auf die aktuelle Apocrypha-Version patchen konnten, dass alle alten Classic-Clients (eine Eve-Variante mit älterer DirectX-Version) ein Upgrade auf den neuen Premium-Client erhielten, und dass wir den aktuellen Apocrypha-Client in Zukunft ebenfalls patchen konnten (dazu nutzten wir simulierte Patch-Versionen). Schließlich mussten wir ebenfalls sichergehen, dass die DVD auf den vielen Apple- und unzähligen Windows-Konfigurationen unserer Spieler funktionierte. Nachdem das alles erledigt war, schickten wir die Masterdisk zu Atari für deren eigene Tests, und die Produktion begannn dann innerhalb der nächsten zwei Monate.

Vorbereitung der virtuellen Welt

Für unser »Virtual World«-Team beginnt die Arbeit mit einem neuen Addon einige Wochen vor der Veröffentlichung, wenn sich der Code der finalen Testphase nähert und wir einen Zeitplan für die Auslieferung erstellen wollen. Wir sehen uns dabei Analysen der Server-Performance mit dem neuen Apocrypha-Code gegen die Performance der gegenwärtig aktuellen Live-Version an und planen entsprechend vorausschauend eventuelle Hardware-Upgrades. Die wichtigsten Ziele für den Tag der Auslieferung sind eine möglichst kurze Offline-Zeit unserer Server sowie die Fähigkeit, alle Änderungen jederzeit rückgängig zu machen, wenn etwas vielleicht doch schiefgehen sollte.

Einzigartig in Apocrypha war das Hinzufügen von 2.500 neuen Sonnensystemen, die den Raum hinter den Wurmlöchern füllen. Für unsere Hardware bedeutete das möglicherweise erheblich höhere Anforderungen. Glücklicherweise ist die Architektur des Server-Clusters hinter Eve sehr flexibel in dieser Hinsicht. Trotzdem verlangte der schiere Umfang der Neuerungen danach, bereits im Vorfeld gründlich zu analysieren, ob wir dazu in der Lage waren. Bei

Im Zuge der Veröffentlichung des Addons Apocrypha gab es Eve Online erstmals seit 2003 auch als **DVD-Version** im Handel.



Käufer der DVD-Version erhalten als Bonus **dieses Shuttle** – auch für einige langjährige Fans Grund genug zum Kauf der Box.

der Planung lag unser Hauptaugenmerk auf der möglichst reibungslosen Spielerfahrung am ersten Tag der Erweiterung, und das ohne Ärgernisse wie serverseitige Lags. Nach dem Start hatten wir dann eine verlässliche Performance-Benchmark, wie sich Apocrypha auf dem Live-Server schlägt, und konnten unsere Hardware-Einteilung entsprechend anpassen.

Während der gesamten Entwicklungszeit von Apocrypha stand das Virtual World Team mit den Programmierern der Wurmlochsysteme in Kontakt und erhielt Schätzungen zum CPU- und RAM-Verbrauch pro System. Zusätzlich versuchten wir, Voraussagen zu treffen, wie viele Spieler die neuen Sonnensysteme auch sofort betreten würden. Nachdem wir einige grundlegende Schätzungen angestellt hatten, rüsteten wir unsere Hardware entsprechend auf. Letzten Endes war es aber dennoch nur eine Hoffnung (wenn auch eine gut geplante), dass die Maßnahmen reichen würden.

Patch-Prüfungen

Nachdem alle Entwicklerteams fertig waren, erstellten wir einen Release Candidate und jagten ihn durch unsere Batterie an Tests auf drei Kontinenten. Dann spielten wir den Build auf unseren Testserver und ließen die Bughunter und regulären Spieler nach Belieben testen. Wir arbeiteten einige Tage fast rund um die Uhr, um alle ernsthaften Probleme zu lösen. Dabei führten wir ständig weitere Regression Tests durch. Als wir die finale Version in Händen hielten, erzeugten wir damit alle Client- und Patch-Versionen. Da wir gegenwärtig ein Single-Patch-System nutzen (wenn Sie sich mit einer beliebigen Eve-Version der letzten Jahre einloggen, reicht ein einziger Patch für alle aus, um auf die aktuelle Version zu kommen; die meisten anderen Spiele nutzen gestaffelte Patch-Systeme), mussten wir insgesamt 40 unterschiedliche Patch-Versionen erstellen (inklusive Windows- und Apple-Versionen sowie manuelle Workarounds für sicherheitskritische Probleme). Jeder Patch wurde lokal getestet und dann zu unserem Hosting-Partner hochgeladen. Da wurden sie erneut getestet, und schließlich ein drittes Mal nach dem Download auf Herz und Nieren geprüft.

Momentan arbeiten wir daran, die Anzahl der Patch-Versionen zu reduzieren, um die Testprozedur weniger aufwändig zu gestalten und sie leichter zu automatisieren – zum Launch von Apocrypha hatten wir aber jeden der 40 Patches dreimal von Hand getestet, mehrere nach gefundenen Dateifehlern (das kann beim Upload-Prozess, beim Verschieben der Dateien auf den Hosting-Servern oder bereits beim Erstellen der Patch-Version passieren) ersetzt und erneut überprüft.

Der Release-Tag

Ein Release-Tag bei CCP ist immer hektisch und aufregend. Bei Apocrypha war es aufgrund des schieren Umfangs, der neuen Grafik-Engine sowie der zahllosen neuen Inhalte und dem gleichzeitigen Erscheinen der DVD-Version sogar noch anspruchsvoller. Trotzdem haben wir große Fortschritte beim Reduzieren der Offline-Zeit unseres Servers gemacht – bei Apocrypha umfasste sie weniger als 13 Stunden, früher waren es meist eher 24 Stunden.

Der ganze Prozess begann damit, dass ein »Virtual World Administrator« und ein »Virtual World Database Administrator« um zwei Uhr nachts im Rechenzentrum eintrafen und sich mit Hilfe von viel Koffein auf eine lange Nacht und das Herunterfahren des Live-Servers »Tranquility« um drei Uhr vorbereiteten. Einer der Hauptgründe für das Verkürzen der Offline-Zeiten war, dass wir von offline durchgeführten Sicherungen unserer Datenbank zu online durchgeführten Backups wechselten, die es uns erlaubt, den gut fünfstündigen Prozess vor dem Herunterfahren der Server durchzuführen.

Am Erscheinungstag von Apocrypha gab es um drei Uhr, als der Cluster herunterfuhr, nur eine halbstündige Wartezeit, bis das Datenbank-Backup fertig war. Dann aktualisierten wir die Server auf die aktuelle Apocrypha-Version, was entgegen der landläufigen Meinung ein sehr schneller und einfacher Vorgang ist. Wenn nur das Verteilen einer Erweiterung wie Apocrypha auch so einfach wäre! Im Detail lief der Prozess zweigeteilt ab – die nötigen Datenbank-Updates wurden nacheinander eingespielt, der Server-Code selbst wurde auf allen, etwa 100 Servern des Clusters gleichzeitig installiert.

Trockentests

Neben den Patch-Tests am Veröffentlichungstag führt unsere Qualitätskontrolle auch »Trockentests« auf dem Live-Server durch. Dabei wird der Server auf den aktuellen Versionsstand gebracht und dann in einem Modus gestartet, der nur Verbindungen von CCP zulässt. Damit wollen wir sicherstellen, dass alles auch auf dem Live-Server gut funktioniert. Denn trotz der zahlreichen vorangegangenen Tests kann jede Veränderung im Umfeld zu unvor-

hersehbaren Problemen führen. Anders als andere MMOs, die auf voneinander getrennte Server setzen, tummeln sich bei Eve alle Spieler weltweit auf einem einzigen Server(-Cluster) – manchmal über 50.000 gleichzeitig. Das macht eine Duplizierung des Servers zu Testzwecken sehr kompliziert. Es ist daher nicht ungewöhnlich, dass sich Gameplay-Aspekte auf dem Live- und dem Test-Server unterscheiden. Schon aus reiner Notwendigkeit ist der Test-Server sehr viel kleiner als der Live-Server, wodurch Diskrepanzen entstehen können.

Die letzten Minuten

Die gesamte Qualitätskontrolle und mehrere andere Gruppen testeten alle neuen Features sowie andere anfällige Bereiche des Spiels. Das ist jedes Mal eine aufregende Zeit ohne einen ruhigen Moment. Da das gesamte QA-Team aus begeisterten Eve-Spielern besteht, die Qualität sehr ernst nehmen, versuchen sie auf jede erdenkliche Art, mögliche Fehler zu entdecken. Was wir den Biloski-Effekt (nach dem QA-Tester Kevin Biloski) nennen, verdeutlicht das: Egal wie gut ein Patch getestet wird, jedes Mal finden wir in den späteren Stadien der Trockentests noch ein kritisches Problem. Größtenteils wird das durch die Kombination von obskuren Spielmechaniken gefunden, auf die nur die am weitesten fortgeschrittenen Eve-Spieler kommen. Häufig können wir aber aufatmen, denn meist hängen diese Fehler mit den Testsystemen oder deren Konfiguration zusammen. Manchmal gibt es aber auch ein echtes Problem. In diesen Fällen muss eine kurzfristige Entscheidung getroffen werden, ob wir den Patch zurücknehmen, bis der Fehler behoben ist. Oder ob wir den Patch freigeben, die Spieler über das Problem sowie mögliche Workarounds informieren und versuchen, es innerhalb der nächsten ein oder zwei Tage aus der Welt zu schaffen. Im Falle von Apocrypha war der kritische Fehler, der laut dem Biloski-Effekt von einem ungenannten Tester gefunden wurde, ein auf den Test-PC beschränktes Problem und kein Software-Defekt. Das war ein Glück für uns, da aufgrund der DVD-Veröffentlichung kein Rollback auf die ältere Eve-Version in Frage kam.

Nachdem die Trockentests fertig, sämtliche Feuer gelöscht und die Pulsschläge aller wieder auf ein Normalmaß gesunken waren, beschlossen wir, dass es an der Zeit war, Apocrypha online gehen zu lassen. Die Qualitätskontrolle gab grünes Licht, die Tore zu öffnen. Um 15:30 (eine halbe Stunde vor dem Zeitplan) änderten wir eine einzige Einstellung auf dem Server, und Spieler aus aller Welt strömten herein, um Wurmlöcher zu suchen und die neuen Sleeper-NPCs zu jagen. Neue Eve-Spieler erlebten ein stark verbessertes Tutorial mit einfacherer und verständlicherer Charaktererstellung. Unsere offiziellen Foren füllten sich mit Berichten über unermessliche Reichtümer und schmerzliche Verluste, ganz wie die Geschichte am Anfang dieses Artikels.

Die nächsten Tage

Nur weil wir Apocrypha erfolgreich zu den Spielern gebracht hatten, war die Arbeit natürlich noch längst nicht getan. Wir bildeten ein Team aus Mitgliedern mehrerer Abteilungen (inklusive Community Relations, Customer Service, Quality Assurance, Virtual World Operations und Eve Software), um das Spiel und die Meinungen der Spieler mehrere Tage nach der Veröffentlichung zu überwachen. Außerdem hatten wir bereits ein paar »Verbesserungswochen« für das Team eingeplant, falls sich doch noch Probleme mit Apocrypha zeigen sollten.

Im Falle von Apocrypha war es vor allem der erhöhte Lag, den Spieler in stark bevölkerten Gegenden des Eve-Universums erlebten. Das Problem lösten wir, indem wir einige Änderungen an der Datenbank durchführten, um die Situation besser überwachen zu können. Danach konnten wir mit einem einzigen Hotfix das Problem beseitigen. Außerdem gab es Schwierigkeiten mit dem neu eingeführten Warteschlangensystem für die Skills der Spieler. Um die zu bereinigen, patchten wir in den folgenden Wochen mehrmals und mussten viel Code umschreiben.

Die Zukunft

Insgesamt waren wir bei CCP zufrieden mit der Auslieferung von Apocrypha, aber wir wollen uns nicht auf Lorbeeren ausruhen. Momentan stecken wir bereits tief in der Entwicklung des nächsten großen Addons. Dafür haben wir unsere Arbeitsprozesse noch einmal genau unter die Lupe genommen und an die neuen Erkenntnisse angepasst. Apocrypha war zwar ungewöhnlich schwierig und kompliziert aus oben genannten Gründen, trotzdem haben wir es schneller und problemloser als alle vorangegangenen Erweiterungen fertiggestellt. Basierend auf diesen Erfahrungen und den vielen daraufhin gestarteten internen Initiativen, blicken wir sehr optimistisch in die Zukunft von Eve Online.

Kurt van Meter, James Wyld

In **großen Flotten** erkundeten die Spieler sofort nach dem Hochfahren des Servers die neuen Gebiete.

Einer der **neuen T3-Kreuzer**, die mit Apocrypha eingeführt wurden.

XNA kann mehr
Shader-Einbindung in XNA

In diesem Artikel beschäftigen wir uns mit der Entwicklung eines universellen Shader-Systems im XNA GameStudio – von der Content-Pipeline-Erweiterung bis hin zum eigentlichen Rendern.

Marc Ueberall

ist Gründer und CEO des BigBlackBlock Game-Studios.

Neben Computerspielen für Windows-PCs und Xbox 360 entwickelt das Unternehmen BigBlack-Block die modulare Game Engine »Elder.Core« für Microsofts XNA-Framework. Bevor das Studio in Kontakt mit XNA kam, wurden zahlreiche Plug-ins und Tools für die quelloffene Engine Ogre3D entwickelt, unter anderem der noch heute genutzten GtkRadiant-Klon Ogre3DRadiant.

WWW
makinggames.de

- XNA-Workshop: Einführung in Spiele-KI

Eine Eigenschaft zeichnet das Microsoft XNA GameStudio besonders aus: die vereinfachte Form der Entwicklung. Denn mit XNA sollen auch Neulinge in kurzer Zeit zu brauchbaren Ergebnissen gelangen. Will man aber das Potenzial von XNA vollends ausschöpfen, ist gerade diese Vereinfachung ein Problem. Besonders in Bezug auf die 3D-Darstellung müssen Programmierer viel Eigenarbeit leisten, da das Microsoft XNA GameStudio keine Unterstützung für komplexe Animationsvorgänge oder Partikelsysteme bietet.

Vorbemerkungen

XNA bietet eine sehr einfache Methode, um 3D-Modelle zu rendern. »Einfach« bedeutet häufig aber auch »unflexibel«. Sieht man sich die meisten XNA-Anwendungen und -Spiele an, fallen besonders zwei Methoden ins Auge – ein eigener Content-Prozessor oder eine manuelle Bearbeitung der Modelle.

In den von Microsoft angebotenen Beispielen für den Einsatz von Shadern ist oft ein spezieller Content-Prozessor nötig, um z.B. Normal Maps oder Reflection Maps nutzen zu können. Theoretisch wäre es möglich, für jede Situation einen speziellen Prozessor zu entwickeln, in der Praxis ist dies aber aus folgenden Gründen nicht ratsam: Hielte man sich an eine derartige Vorgehensweise, würde man schnell bei einer unübersichtlichen Anzahl von Modell-Prozessoren enden. Dem Entwickler fiele außerdem die Aufgabe zu, jedem einzelnen Asset den entsprechenden Prozessor zuzuordnen. Schließlich tragen die angesprochenen Beispiele den Shader auf das komplette Modell auf – eine Differenzierung von unterschiedlichen Materialien in einem Modell wäre nicht möglich.

Die andere Methode kommt in dem vorhergehenden Entwicklungsschritt zum Zuge. Hier werden dem Modell direkt in der 3D-Modellierungsanwendung die zu verwendende .fx-Shader-Datei und die Werte der Parameter bzw. die einzubindenden Texturen bereitgestellt. Die Material-Daten werden dann beim Export z.B. in das DirectX-Format .x direkt fest in das Modell eingebettet. Benötigt man verschiedene 3D-Modelle, die denselben Shader nutzen sollen, müssen bei späteren Feinabstimmungen oder

Die Shader für diesen **Roboter** und die im Artikel abgebildeten Spiele wurden unter XNA mit den hier vorgestellten Techniken programmiert.

Änderungen des Shaders alle betroffenen Modelle erneut exportiert oder – sofern ein Ascii-Format gewählt wurde – manuell nachbearbeitet werden. Zudem unterstützt nicht jede 3D-Modellierungsanwendung diese Möglichkeit. Die im Folgenden erläuterte Technik ist bei weitem nicht die einzige Lösung des Problems und sollte daher nicht als Dogma aufgefasst werden.

Material Bibliotheks Script

Um Shader beziehungsweise Materialien so flexibel wie möglich in die Content Pipeline einbinden zu können, vergibt man in der 3D-Modellierungsapplikation lediglich einen eindeutigen Namen für das Material und stellt die ge-

```
<?xml version="1.0" encoding="utf-8" ?>
<materials>
    <material name="gateIndicator_ncl1_2" effect="Effects\\beam.fx" technique="1" queue="3" flags="NO_COLLISION" state="TRANSLUCENT">
        <param type="float" name="alphaScale" value="0.5" />
        <param type="texture" name="diffuseMap" value="Textures\\gateIndicator_d.png" />
    </material>
    <region name="tiles">
        <material name="gateBody" effect="Effects\\standard.fx" technique="1" queue="1" flags="IS_SHADOWCASTER|IS_SHADOWRECEIVER" state="ALPHA">
            <param type="float" name="specularAmount" value="0.25" />
            <param type="float" name="specularShininess" value="50" />
            <param type="float" name="bumpAmount" value="0.125" />
            <param type="texture" name="diffuseMap" value="Textures\\gateBody_d.tga" />
            <param type="texture" name="bumpMap" value="Textures\\gateBody_n.tga" />
            <param type="texture" name="specularMap" value="Textures\\gateBody_s.tga" />
        </material>
        <material name="gateFinish" effect="Effects\\screen.fx" technique="1" queue="1" flags="IS_SHADOWCASTER|IS_SHADOWRECEIVER">
            <param type="texture" name="diffuseMap" value="Textures\\gateFinish_d.tga" />
        </material>
        <material name="gateCheckpoint" effect="Effects\\screen.fx" technique="1" queue="1" flags="IS_SHADOWCASTER|IS_SHADOWRECEIVER">
            <param type="texture" name="diffuseMap" value="Textures\\gateCheckpoint_d.tga" />
        </material>
        <material name="gateStart" effect="Effects\\screen.fx" technique="1" queue="1" flags="IS_SHADOWCASTER|IS_SHADOWRECEIVER">
            <param type="texture" name="diffuseMap" value="Textures\\gateStart_d.tga" />
        </material>
    </region>
</materials>
```

Beispiel eines **Material Scripts** im XML-Format.

naue Definition als externe Datei bereit. Da das .net-Framework eine bequeme Lösung zur Verarbeitung von XML-Daten bietet, liegt es natürlich nahe, die Materialbibliothek als XML-Dokument anzulegen.

Die Struktur der XML-Daten kann flexibel ausfallen und unterliegt für die hier angesprochene Technik nur wenigen Vorschriften. Entweder kann für jedes Material eine gesonderte Datei erstellt werden, oder es werden pro Datei Gruppen von Materialien zusammengefasst. Jedes XML-Element, welches ein Material beschreibt, ist mit verschiedenen Attributen belegt. Um das Material eindeutig identifizieren zu können, muss dem Element ein Name angehängt werden. Das Attribut *effect* zeigt an, welche .fx-Datei verwendet werden soll. Ferner sollte noch die Shader Technique als Index oder Name vermerkt werden. Falls die .fx-Datei verschiedene Shader-Modelle verwendet, sollte man auf die Angabe des Namens zurückgreifen, weil so zu Laufzeit durch Anhängen eines Postfix wie z.B. »_2_0« oder »_3_0« die jeweilige Konfiguration angesprochen werden kann. Eine Erweiterung der XML-Attribute ist denkbar; so kann *queue* die Position in der Renderqueue angeben, um z.B. beim Rendern den Opaque-Pass vom Translucent-Pass zu trennen – je höher der Index, desto später wird das Material gerendert. Ein Material-Flag ordnet jedem Material noch grobe Eigenschaften zu wie:

- LightCaster
- LightReceiver
- ShadowCaster
- ShadowReceiver
- NoCollision
- Invisible
- Billboard

Mit dem **state**-Attribut können intern festgelegte RenderState-Konfigurationen angesprochen werden. Als eine Folge von **param**-Elementen werden nun alle relevanten Parameter des Materials angegeben. Da Parameter verschiedene Typen annehmen können, sollte auch das **param**-Element durch verschiedene Attribute beschrieben werden. Hier gibt **name** den Namen des Parameters an, wie er in dem zugrunde liegenden Shader definiert wurde. Das Attribut **type** gibt dabei den zu verwendenden Variablen-Typen an, der den in **value** gesetzten Wert beschreibt. Einige mögliche Enumerationen wären:

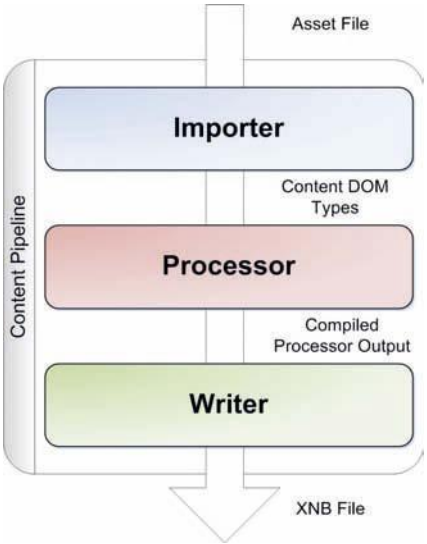
- bool
- int
- float
- double
- vector2
- vector3
- vector4
- color
- texture

Wichtig: Sie sollten nur die Parameter beschreiben, die nicht dynamischer Natur sind, zum Beispiel Transformations- und Projektionsmatrizen und Frame-Informationen; also alle Parameter, die durch **Standard Annotation Semantics** (SAS) beschrieben sind. Für eine noch weitergehende Verwendung der Materialbibliothek können beispielsweise neue XML-Elemente die physikalischen Eigenschaften beschreiben.

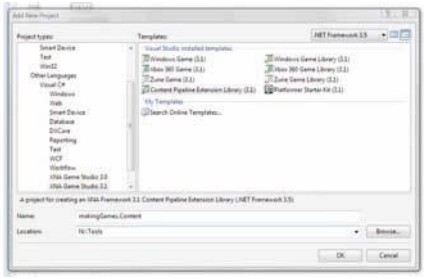
Anpassung der Content Pipeline

Um das Materialbibliothek-Script nutzen zu können, müssen Erweiterungen an der Content Pipeline vorgenommen werden. Dabei ist die Content Pipeline von XNA grundsätzlich in drei Phasen eingeteilt.

Der Content Importer liest die Daten aus den jeweiligen Asset-Dateien und übergibt diese als internes Datenbündel (Content DOM Type), welches der Content Processor verarbeitet und die Daten kompiliert. Dieser Output wird an den Content Type Writer gesendet, der die Daten in das XNB-Format schreibt. Jede der drei Phasen greift auf verschiedene Klassen innerhalb der Content Pipeline zu; so verwendet der Model-Processor den Material Processor, der Material Processor den Texture Processor und so weiter.



Schematischer Ablauf der Kompilierung eines Assets durch die **XNA Content Pipeline**.



Erzeugung eines »**Content Pipeline Extension Library**«-Projektes.

Ebenso besteht ein Content Type Writer aus verschiedenen spezialisierten Klassen, um jeweilige Content Types zu schreiben. Egal um welche Art von Asset es sich handelt: Der hier beschriebene Ablauf ist generell gleich.

Aus logistischen Gründen empfiehlt es sich, Content Pipeline und Runtime-Komponenten zu trennen und in verschiedene Assemblies zu betten, da z.B. bei jedem Codechange auch die Neukompilierung aller Assets fällig wird. Hierzu wird zunächst ein neues »Content Pipeline Extension Library« -Projekt in VisualStudio angelegt.

Zunächst wird die Hilfsklasse **MGMaterialLibrary** erstellt. Diese importiert alle Dateien mit der Endung **.material* aus dem Verzeichnis *»Materials«* (ein Unterverzeichnis des Content-Projektes der jeweiligen Anwendung). Alle Daten werden in einem Dictionary gesammelt und für den späteren Gebrauch per Indexer bereitgestellt.

Im nächsten Schritt muss ein neuer Material Processor erstellt werden, der im späteren Verlauf vom Model Processor aufgerufen wird. Der Material Processor wird mit einer **Process**-Methode angesprochen, wodurch der Methode eine Information über das derzeitige Material als Parameter gegeben wird. Hier wird der Name des im Modells benutzten Materials in der von **MGMaterialLibrary** verwalteten Instanzen gesucht und verarbeitet.

XNA verwendet die Klasse **EffectMaterialContent**, um den Geometriedaten mitzuteilen, welcher Shader mit welchen Parametern verwendet werden soll. Diese Klasse wird mit dem Namen des einzubindenden Shaders initialisiert und mit allen relevanten Daten gefüttert. Zuerst werden die generischen Informationen über das Material in das **OpaqueDataDictionary** gespeichert. Um diese Daten später besser identifizieren zu können, werden die jeweiligen Namen mit Unterstrichen als Pre- und Postfix versehen. Der Material Processor durchläuft anschließend die in der internen Materialinstanz gesetzten Parameter und sortiert diese den Feldern der Klasse **EffectMaterialContent** zu. Verwendete Texturen werden in einer Liste, alle anderen Shader-Parameter in dem **OpaqueDataDictionary** gespeichert. Der eigentliche Kompiliervorgang wird von der Basisklasse **MaterialProcessor** übernommen.

Da nun die Grundlage für die Verarbeitung der Materialien geschaffen ist, muss ein neuer Model Processor erstellt werden. Dieser hat die Aufgabe, die Geometriedaten zu kompilieren und mit sämtlichen Materialien zu verknüpfen.

Die Basisklasse **ModelProcessor** stellt die Methode **ConvertMaterial** zur Verfügung, die aus dem im 3D-Asset angegebenen Material eine interne Datenstruktur erstellt. Hier wird nun der neue Material-Processor eingebunden, um aus den XML-Daten ein XNA-Effect-Material zu generieren. Die Methode **Process** ist auch hier der wichtigste Bestandteil dieses Prozessors. Da

Der **Model-Prozessor**.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Linq;
using System.Text;

using Microsoft.Xna.Framework;
using Microsoft.Xna.Framework.Graphics;
using Microsoft.Xna.Framework.Content.Pipeline;
using Microsoft.Xna.Framework.Content.Pipeline.Graphics;
using Microsoft.Xna.Framework.Content.Pipeline.Processors;

namespace makingGames.Content
{
    [ContentProcessor(DisplayName = "Model - makingGames")]
    public class MGModelProcessor : ModelProcessor
    {
        #region Properties

        private OpaqueDataDictionary Parameters
        {
            get
            {
                OpaqueDataDictionary param = new OpaqueDataDictionary();
                param["ColorKeyColor"] = this.ColorKeyColor;
                param["ColorKeyEnabled"] = this.ColorKeyEnabled;
                param["GenerateMipmaps"] = this.GenerateMipmaps;
                param["ResizeTexturesToPowerOfTwo"] = this.ResizeTexturesToPowerOfTwo;
                param["TextureFormat"] = this.TextureFormat;
                return param;
            }
        }

        #endregion

        #region ModelProcessor Members

        protected override MaterialContent ConvertMaterial(MaterialContent material, ContentProcessorContext context)
        {
            return context.Convert<MaterialContent, MaterialContent>(material, "MGMaterialProcessor", this.Parameters);
        }

        public override ModelContent Process(NodeContent content, ContentProcessorContext context)
        {
            ModelContent model = base.Process(content, context);
            for (int i = 0; i < model.Meshes.Count; i++)
            {
                for (int j = 0; j < model.Meshes[i].MeshParts.Count; j++)
                {
                    model.Meshes[i].MeshParts[j].Tag = new MGMaterialTag(model.Meshes[i].MeshParts[j].Material.OpaqueData);
                }
            }
            return model;
        }

        #endregion
    }
}
```

Der **Material-Prozessor** und die Hilfsklassen.

```
[ContentTypeWriter]
public class MGMaterialTagWriter : ContentTypeWriter<MGMaterialTag>
{
    protected override void Write(ContentWriter output, MGMaterialTag value)
    {
        output.Write(value.Name);
        output.Write(value.State);
        output.Write(value.Flags);
        output.Write(value.Technique);
        output.Write(value.Queue);
    }

    public override string GetRuntimeType(TargetPlatform target)
    {
        return typeof(MGMaterialTag).AssemblyQualifiedName;
    }

    public override string GetRuntimeReader(TargetPlatform target)
    {
        if (target == TargetPlatform.Windows)
        {
            return "makingGames.MaterialReader, makingGamesSample, Version=1.0, Culture=neutral";
        }
        else if (target == TargetPlatform.Xbox360)
        {
            return "makingGames.MaterialReader, makingGamesSample, Version=1.0, Culture=neutral";
        }
        else if (target == TargetPlatform.Zune)
        {
            return "makingGames.MaterialReader, makingGamesSample, Version=1.0, Culture=neutral";
        }
        return "makingGames.MaterialReader, makingGamesSample, Version=1.0, Culture=neutral";
    }
}

[ContentProcessor(DisplayName = "Material - makingGames")]
public class MGMaterialProcessor : MaterialProcessor
{
    public override MaterialContent Process(MaterialContent content, ContentProcessorContext context)
    {
        MGMaterialLibrary.Instance.Context = context;
        if (!MGMaterialLibrary.Instance.Contains(content.Name))
        {
            throw new InvalidContentException(string.Format("Material '{0}' not found", content.Name),
                content.Identity);
        }
        MGMaterialEntry m = MGMaterialLibrary.Instance[content.Name];
        if (m == null)
        {
            m = new MGMaterialEntry("generic", content.OpaqueData["effect"] as string, string.Empty,
                string.Empty, string.Empty, string.Empty);
            content.OpaqueData.Remove("effect");
        }
        if (m != null)
        {
            if (m.MaterialFile != "generic") { context.AddDependency(m.MaterialFile); }
            EffectMaterialContent material = new EffectMaterialContent();
            material.Effect = new ExternalReference<EffectContent>(m.EffectFile);
            material.OpaqueData.Add("__name__", content.Name);
            material.OpaqueData.Add("__technique__", m.TechniqueId);
            material.OpaqueData.Add("__queue__", m.QueueId);
            material.OpaqueData.Add("__flags__", m.Flags);
            material.OpaqueData.Add("__state__", m.State);
            foreach (string n in m.Parameters.Keys)
            {
                MGMaterialParameter p = m.Parameters[n];
                if (p.Type == MGMaterialParameterType.Texture)
                {
                    material.Textures.Add(n, new ExternalReference<TextureContent>((string)p.Value));
                }
                else { material.OpaqueData.Add(n, p.Value); }
            }
            m.Tag = new MGMaterialTag(material);
            return base.Process(material, context);
        }
        return base.Process(content, context);
    }
}
```

nichts an der Art der Geometrieverarbeitung geändert werden soll, übernimmt diesen Teil die Basisklasse **ModelProcessor**. Nachdem der erstellte Content von der Basisklasse zurückgegeben wird, hängt der Prozessor die Materialinformationen in das **Tag**-Feld eines jeden Mesh-Parts an. Ein **Tag** wird von jeder Klasse bereitgestellt, die von **ContentItem** abgeleitet ist. Grundlegend ist es möglich, jede Art von Daten in dieses Feld zu schreiben. Dies erfordert aber auch die Erstellung eines spezialisierten Content Type Writers bzw. zur Laufzeit eines Content Type Readers. Seit dem Release von XNA GameStudio 3.1 (Mitte Juni 2009) werden die Datentypen jedoch per Reflection gelesen. Es ist also nicht mehr zwingend erforderlich, die Content Type Writer und Content Type Reader zu erstellen. Trotzdem ist es in einigen Fällen nötig, da aufgrund von verschiedenen Obfuscation-Methoden ein Zugriff per Reflection nicht mehr funktioniert und somit der Content nicht gelesen werden kann.

Jeder Content Type Writer unterliegt mehreren Vorschriften, die zwingend einzuhalten sind. So müssen in den Methoden **GetRuntimeType** und **GetRuntimeReader** genaue und eindeutige Angaben über den Typ des Contents und der Lokalität des Content-Type-Readers gemacht werden. Für die Methode GetRuntimeReader gilt als Rückgabewert die Folge:

```
[fullname], [assemblyname], Version=[version],
Culture=[culture]
```

Diese Informationen werden in die XNB-Datei eingebettet, um der Anwendung mitzuteilen, wo der entsprechende Reader der in **GetRuntimeType** beschriebenen Datenstruktur zu finden ist. Es ist möglich, anhand des Methoden-Parameters **target** festzustellen, ob der Content für die Windows-, Xbox 360- oder Zune-Plattform kompiliert wird. So können leicht plattformspezifische Szenarien vorgegeben werden. In der Methode **Write** wird die eigentliche Arbeit des Writers verrichtet. Sie ist wieder für alle XNA-Plattformen gleich. Die Parameter **writer** und **value** geben den Stream und die derzeitigen Daten an, die geschrieben werden sollen. XNA bietet in der **ContentWriter**-Klasse Support für fast alle Standardtypen, die das .net-Framework und XNA selbst verwenden. Wird ein generisches Objekt geschrieben – wie es bei den Tag-Feldern der Klasse **ContentItem** üblich ist – sind weitere **ContentWriter**-Spezialisierungen nötig. Da es sich bei dem Material lediglich um Werte des Typs **string** bzw. **int** handelt, kommt man dort ohne weitere Arbeit aus.

Um die neuen Funktionen der Content Pipeline nutzen zu können, wird die Pipeline Assembly in das Unterprojekt »Content« der Applikation als Referenz eingefügt. Von nun an stehen im Property Editor des Content-Projektes zwei neue Prozessorarten zur Verfügung.

- Model - makingGames
- Material - makingGames

Der Material Processor ist nicht direkt zu verwenden. Zwar kann man die Script-Dateien der Materialien in das Projekt einfügen, muss diese aber im Property-Editor als nicht zu kompilieren und nicht zu kopieren vermerken. Jedes 3D-Asset wird nun auf den neuen ModelProcessor umgestellt.

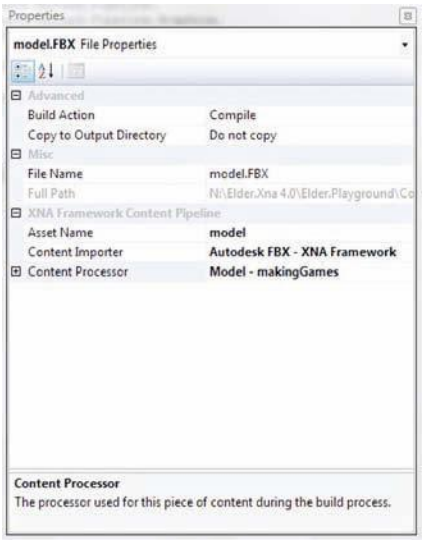
Wird das Projekt gestartet, kompiliert die Content Pipeline die Assets neu und trägt die Materialien entsprechend der Materialbibliothek auf. Ändert man ein Material-Script, so hat auch dies einen erneuten Kompilierungsvorgang der von diesem Material betroffenen Assets zur Folge.

Verwendung der Daten zur Laufzeit

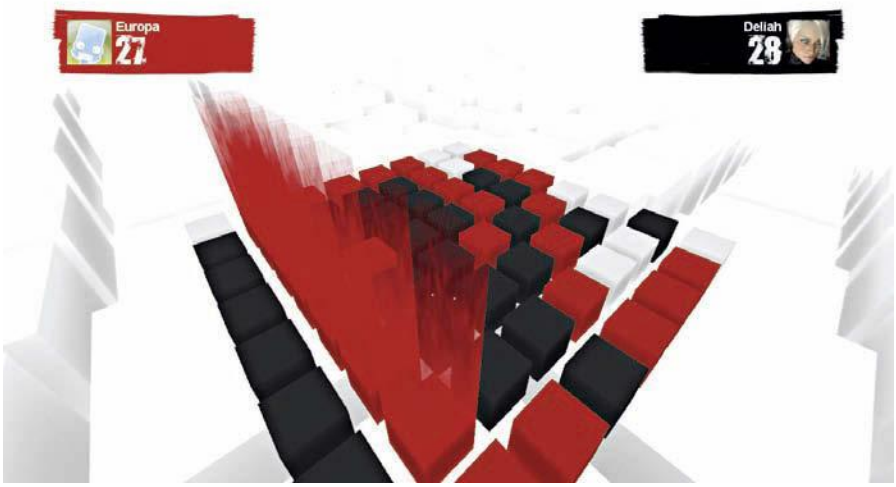
Um den Inhalt des in dem Modell verwendeten Material-Tags nutzen zu können, muss ein Material-Objekt erstellt und der Content Type Reader implementiert werden. Wie schon erläutert ist der Content Type Reader das Gegenstück zum – in der Content Pipeline – verwendeten Content Type Writer. Die Methode **Read** der Basisklasse **ContentTypeReader** wird der Struktur der Daten, wie sie im Content Type Writer ausgegeben wurden, angepasst.

Wird ein 3D-Modell in der XNA-Anwendung mithilfe einer Instanz der Klasse **ContentManager** gelesen, können sämtliche Instanzen von **ModelMeshPart** des Objektes iterieren und das **Tag**-Feld zum Typ **MGMaterial** gecastet werden.

Alle relevanten Informationen stehen somit der Anwendung zur Verfügung. Diese gilt es jetzt möglichst effektiv zu verarbeiten. Wie angesprochen sollen alle mit SAS verbundenen Shader-Parameter automatisch gesetzt werden. Die meisten SAS-Definitionen beziehen sich auf Transformationen, Projektionen und Frame-Informationen. Um jedem Shader bzw. Effect die passenden Daten zu liefern, empfiehlt es sich, eine Hilfsklasse einzuführen, die diese zu jeder Zeit liefern kann. Die Klasse **RenderContext** löst diese Aufgabe und kann sogar zu einer Schlüsselstelle des Render-Prozesses gemacht werden, um z.B. als Singleton-Implementierung die Render States und Render Targets zu managen. Für die hier angesprochenen Zwecke sollte die Klasse die verwendete Kamera registrieren,



Einstellungen im **Property-Editor.**



Das Brettspiel **Brixx** verwendet die hier vorgestellten Shader-Techniken und soll demnächst sowohl für den PC als auch für die Xbox 360 erscheinen.

aktuelle Frame-Informationen speichern (Frame-Nummer, Zeitindex und mehr), derzeitige Environment und Shadow Maps aufnehmen und globale Szeneninformationen wie zum Beispiel die Farbe des Umgebungslichts oder Daten bezüglich des Fogs sammeln. Vermieden werden sollten Daten, die gänzlich unnötig sind oder nur selten gebraucht werden. Stehen die Daten zur Verfügung, wird eine weitere Hilfsklasse benötigt. Eine Render-Queue soll die Aufgabe der Darstellung von 3D-Modellen übernehmen.

Render-Queue

Die Render-Queue sammelt alle Geometrie-Teile, die das gleiche Material benutzen, und arbeitet sie nach und nach ab. So wird unnötiges Wechseln zwischen verschiedenen Effekten vermieden. Aus Gründen der Komplexität wird die Arbeit der Render-Queue nur schematisch beschrieben, eine Implementierung sollte keine großen Probleme bereiten.

Quellcode der **Laufzeit-Komponenten**.

```
public enum SemanticId
{
    Unknown,
    CameraFar,
    CameraNear,
    /* Weitere Semantics */
}

public class SASShader : IDisposable
{
    #region Fields

    private string m_technique = string.Empty;
    private Effect m_effect = null;
    private List<ShaderParameter> m_lightParameters = new List<ShaderParameter>();
    private List<ShaderParameter> m_worldParameters = new List<ShaderParameter>();
    private List<ShaderParameter> m_viewParameters = new List<ShaderParameter>();
    private List<ShaderParameter> m_globalParameters = new List<ShaderParameter>();

    #endregion

    #region Constructors

    public SASShader(Effect effect)
    {
        this.m_effect = effect;
        this.CacheParameters();
    }

    #endregion

    #region Methods

    private void CacheParameters()
    {
        if (this.m_effect == null)
        {
            throw new ShaderException(Messages.SH001);
        }
        for (int i = 0; i < this.m_effect.Parameters.Count; i++)
        {
            string name = this.m_effect.Parameters[i].Name.ToLower();
            if (!string.IsNullOrEmpty(this.m_effect.Parameters[i].Semantic))
            {
                ShaderParameter p = new ShaderParameter(this.m_effect.Parameters[i]);
                try
                {
                    p.Id = ((SemanticId)Enum.Parse(typeof(SemanticId), this.m_effect.Parameters[i].Semantic, true));
                }
                catch { p.Id = SemanticId.Unknown; }
                switch (p.Id)
                {
                    case SemanticId.CameraFar:
                    case SemanticId.CameraNear: this.m_viewParameters.Add(p); break;
                    /* Weitere Semantics */
                    default: this.m_globalParameters.Add(p); break;
                }
            }
        }
    }

    internal void SetViewParameters()
    {
        GraphicManager context = GameManager.Instance.GraphicManager;
        for (int i = 0; i < this.m_viewParameters.Count; i++)
        {
            ShaderParameter param = this.m_viewParameters[i];
            switch (param.Id)
            {
                case SemanticId.CameraFar: param.Parameter.SetValue(context.Camera.Far); break;
                case SemanticId.CameraNear: param.Parameter.SetValue(context.Camera.Near); break;
                /* Weitere Semantics */
                default: throw new ShaderException(string.Format(Messages.SH002, param.Parameter.Semantic), param);
            }
        }
    }

    /* ... */

    #endregion
}
```

Bevor ein Modell gerendert werden kann, muss es sich bei der Queue anmelden; dieser Schritt sollte nur einmal und zwar nach dem Laden des Content Assets (etwa in einer Methode wie **LoadContent**) erfolgen. Die Queue sortiert jedes Teil des Modells dem verwendeten Material und seinem Gruppen-Index (Position innerhalb der Queue) zu und gibt anschließend einen Verweis auf den Queue-Eintrag zurück. Über Listen und Dictionaries werden diese Einträge strukturiert, wobei nach folgender Regel vorgegangen wird:

1. Sortieren nach *MeshPerGroup*
2. Sortieren nach *MeshPerEffectMaterial*
3. Sortieren nach *MeshPerTechnique*
4. Speichern als *MeshInstance*

Die Differenzierung der einzelnen Objekte kann generell über den Hash-Wert des entsprechenden Objektes erfolgen.

Wird ein Modell mehrfach verwendet, registriert die Queue dieses anhand des Hash-Wertes des Geometrie-Objektes und gibt lediglich den vorherigen Eintrag wieder. Eine Wrapper-Klasse, welche die XNA-Klasse **Model** kapselt, kann nun diese Einträge sammeln und speichern. Durch die Methode **Draw** der Wrapper-Klasse wird der Render-Befehl gegeben, und sie meldet der Render-Queue nur die zu verwendende Transformationsmatrix für die Modell-Instanz und z.B. die – für den Fall eines Forward Renderers – involvierten Lichtquellen. Jeder Eintrag der Render-Queue muss in der Lage sein, die erforderlichen Daten pro Instanz zu speichern. Ist der eigentliche Render-Vorgang abgeschlossen, wird der Render-Queue mitgeteilt, dass sie jetzt ihren Inhalt flushen soll. Die Queue beginnt bei dem niedrigsten Gruppen-Index und arbeitet sich so über die verschiedenen Sortierungskriterien bis zu den einzelnen Instanzen der Geometrie-Einträge vor. Bei Erreichen eines neuen Effektes wird dieser einmalig (pro Frame) eingebunden und seine Parameter nur noch in den nächsten Schritten gemäß der SAS-Definitionen auf den neuesten Stand gebracht. Ist der jeweilige Geometrie-Eintrag erreicht, werden alle Instanzen nacheinander abgearbeitet. Hier werden die Transformationsmatrix und eventuelle Lichtquellen an den Shader-Effekt übergeben und der Vertex- und Indexbuffer des Eintrags einmal pro Instanz eingebunden und gerendert. Damit die an SAS gebundenen Werte auch richtig gesetzt werden können, muss eine weitere Wrapper-Klasse eingeführt werden – die Klasse **SASEffect**.

Parameter

Die Klasse **SASEffect** nimmt bei Initialisierung die jeweilige XNA-**Effect**-Instanz auf und verarbeitet deren Daten. Genauer gesagt werden die von der Klasse **Effect** angebotenen Informationen der Parameter-Semantics genutzt, um Laufzeitwerte zuordnen zu können.

In dem aufgeführten Beispiel wird ein Großteil der in SAS definierten Semantiken genutzt und um eigene erweitert. Um die Effizienz des

Systems zu steigern, werden alle Parameter in vier verschiedene Parameter-Listen unterteilt:

- View-Parameter: enthält alle Parameter, die zur Projektion oder Beschreibung von Kameradaten dienen.
- Light-Parameter: enthält alle Parameter, die zur Beleuchtung der Szene benötigt werden.
- World-Parameter: enthält alle Parameter, die zur Transformation der Geometrie beitragen.
- Global-Parameter: enthält alle anderen Parameter, die nicht in obige Listen gefallen sind.

Die Methode **CacheParameters**, die von dem Konstruktor aufgerufen wird, iteriert den Inhalt des Feldes **Parameters** und erzeugt aus jedem Eintrag ein **SASParameter**-Objekt. Da ein String-Vergleich zur Laufzeit zu langsam wäre, wird eine Enumeration der SAS-Definitionen verwendet. Hierzu wird der String-Wert des Feldes **Semantic** eines jeden Effektparameters mithilfe der **Enum**-Method **Parse** zum Typ **SemanticType** gecastet. Anschließend werden die entsprechenden Parameter gemäß ihrem Typ den vier Parameter-Listen hinzugefügt.

Entsprechend der vier Parameter-Listen stellt die Klasse **SASEffect** vier Methoden zur Verfügung, die durch die Render-Queue aufgerufen werden. Der Inhalt der jeweiligen Liste wird iteriert und jedem beinhalteten Parameter gemäß seiner Semantik der derzeitige, in der Klasse **RenderContext** hinterlegte oder per Methoden-Parameter übergebene Wert zugeordnet. Alle vier Methoden arbeiten nahezu analog. Die Queue trägt während des Render-Vorgangs nicht alle Shader-Parameter gleichzeitig auf, sondern bedient sich jeder der vier oben angesprochenen Methoden. **SetViewParameters** und **SetGlobalParameters** wird nur einmal pro Effekt, **SetLightParameters** und **SetWorldParameters** einmal pro Modell-Instanz aufgerufen. So wird vermieden, dass überflüssige Informationen zwischen Soft- und Hardware ausgetauscht werden, da sich die Global- und View-Parameter in der Regel während eines Rendercalls nicht ändern.

Erweiterungsmöglichkeiten

SAS-Befehle können auf sehr einfache Art erweitert werden. Als Beispiel soll hier die Einführung einiger neuer Semantiken dienen, die die Repräsentation einer Sonne bilden. Die Klasse **RenderContext** erhält ein neues Feld **Sun**, das die »globale« Lichtquelle aufnimmt. Durch Einführung neuer Einträge in der Enumeration **SemanticId** werden die Semantiken **SunDirection**, **SunIntensity** und **SunColor** bereitgestellt und in der Methode **CacheParameters** der Klasse **SASEffekt** in die Liste der Light-Parameter eingetragen. Durch Einfügen der neuen Semantiken in die **switch**-Anweisung der Methode **SetLightParameters** und der Vergabe der Werte durch die Klasse **RenderContext** stehen diese nun dem System zur Verfügung und können im Shader-Code durch entsprechende Deklaration verwendet werden.

Erweiterungen können oder müssen nicht nur an dem SAS-System vorgenommen werden. Eine Übergabe von Callback-Methoden durch die Registrierung der Modell-Instanzen bei der Render-Queue kann in einigen Fällen einen sinnvollen Zusatz darstellen, da so Event-basiert Shader-Parameter kurz- oder langfristig geändert werden können.

Werden im Shader verschiedene Konfigurationen der Shader Models verwendet, kann dies ebenfalls als Erweiterung in das hier beschriebene System eingebaut werden. Dann müssen allerdings Konventionen bezüglich der Namensgebung getroffen und strikt eingehalten werden. Die Shader Techniques sollten beispielsweise als **DiffuseNormalSpecular_2_0** und **DiffuseNormalSpecular_3_0** benannt und im Material-Script lediglich als **DiffuseNormalSpecular** bezeichnet werden. Wird das 3D-Modell und der entsprechende Material-Eintrag von der Anwendung geladen, kann an dieser Stelle die Systemkonfiguration und somit die Shader-Model-Kapazität getestet werden. Steht SM3.0 zur Verfügung, wird dem Feld **Technique** des Material-Tags ein »_3_0«, andernfalls ein »_2_0« angehägt. Stellt XNA in Zukunft auch SM4.0 bzw. SM5.0 bereit, können diese Konfigurationen natürlich ebenfalls angehängt werden.

Marc Ueberall

Die Marble-Madness-Hommage **HiRollerz** wurde mit dem XNA Framework 3.0 entwickelt und verwendet DirectX 9.0c (Shader Model 3).

Das Live Aid der Spielebranche

Gründung eines Charity-Publishers

Einmal ganz ohne Deadline und enge Zeitpläne entwickeln? Keine Wunschvorstellung, sondern die Realität beim weltweit ersten gemeinnützigen Publisher der Spieleindustrie. Gründer Martin de Ronde erzählt uns die Entstehungsgeschichte des außergewöhnlichen Projekts OneBigGame.

Martin de Ronde

ist Gründer und Vorsitzender des Publishers OneBigGame.

Martin begann seine Karriere vor zwölf Jahren als PR- und Development Manager bei einem Spiele-Publisher. Kurze Zeit später gründete er sein erstes eigenes Studio, das er 1999 an das Multimedia-Unternehmen Lost Boys verkaufte. Nach der Umbenennung von Lost Boys in Guerilla Games war er als kaufmännischer Leiter für Killzone und Shellshock: Nam '67 verantwortlich. 2006 gründete de Ronde OneBigGame in San Francisco. Neben dieser ehrenamtlichen Tätigkeit arbeitet er als Consultant und als Geschäftsführer seines neuen Entwicklerstudios Karami.
martin.de.ronde@onebiggame.org

OneBigGame ist der erste gemeinnützige Publisher in der Spiele-Industrie. Seit ich ihn 2006 gemeinsam mit diversen Branchen-Veteranen aus Wohltätigkeitszwecken gegründet habe, ist unser erklärtes Ziel die Veröffentlichung von Computerspielen – finanziert und entwickelt von den bekanntesten Köpfen der Industrie. Alle Erlöse kommen Kinderhilfsorganisationen auf der ganzen Welt zugute. Unser Programm startet Ende des Jahres mit einer Vielzahl hochkreativer Spiele, erdacht und umgesetzt sowohl von bekannten Namen, als auch von etablierten Indie-Entwicklern.

Wie alles begann

OneBigGame lässt sich wohl am besten mit Live Aid vergleichen, der berühmten Charity-Initiative der Musikindustrie in den 80ern. Zahlreiche Stars sind damals zusammengekommen, um Hit-Singles aufzunehmen, deren Erlöse den Opfern der Hungersnot in Äthiopien zugutekommen sollten. Sowohl »Do They Know It's Christmas Time« als auch »We Are The World« waren nicht nur Riesenerfolge, sondern schärften gleichzeitig den Blick für die aussichtslose Lage, in der sich damals Millionen von Menschen befanden. Die Singles begleiteten eine Reihe von Konzerten rund um den Globus. Wieder mit den ganz Großen des Musikbusiness, die ihre berühmtesten Songs spielten und so Unsummen an Geld für Bedürftige in Afrika einbrachten.

Es war diese große Wohltätigkeitsveranstaltung, die als Inspiration für OneBigGame diente. 2005 sah ich eine Dokumentation zum 20. Geburtstag von Live Aid und Band Aid und fragte mich, ob so ein Großprojekt nicht auch in der Spiele-Industrie zu realisieren wäre. Mal ehrlich, wenn Musik für die Popkultur in den 90ern steht, ist dann nicht Gaming die treibende Kraft der heutigen Unterhaltungsindustrie? Und ganz abgesehen davon: Warum unternimmt eine so große Industrie wie die Spielebranche mit einem Wert von mehr als 50 Milliarden Dollar so wenig für wohltätige Zwecke, während Musik-, Kino- und Fernsehstars andauernd in alle möglichen Variationen des Spendensammelns eingebunden werden?

Die ursprüngliche Idee

Ein Argument könnte sein, dass die Spielebranche nicht solche internationalen und glamourösen Stars hervorbringt, wie die zuvor erwähnten Industrien. Unsere Antwort darauf: Wir haben nicht nur Berühmtheiten mit hohem Erkennungswert in den Spielcharakteren selbst, sondern unsere Branche bringt auch berühmte Designer und Entwickler mit Star-Appeal hervor. Wie viele Fans hat Blizzard? Und wie viele Millionen Menschen kennen »Die Sims« und ihren Schöpfer Will Wright? Wenn wir diese Star-Power für gemeinnützige Zwecke einsetzen würden, dazu noch über die Entwicklung innovativer Spiele vergleichbar mit der Band-Aid-Hit-Single, dann könnten wir die Welt tatsächlich ein Stückchen besser machen. Und so begann unsere verrückte, ambitionierte Reise.

Die ursprüngliche Idee war sehr einfach: Lasst uns ein großes, neues Next-Gen-Spiel (daher unser Name) auf den Markt bringen, das wohltätigen Zwecken dient, von Star-Designern entwickelt wird und dessen Erlös Kinderhilfsprogrammen in aller Welt zugutekommt. Wir baten zunächst eine Handvoll weniger bekannte Designer, ein »Blueprint« für dieses große Spiel zu entwickeln, und heraus kam letztlich das Format eines »Hub Games«, das alle Arten von Mini-Games in sich vereint. Diese Spiele wiederum sollten allesamt von Star-Designern entwickelt werden, die einen immensen Bekanntheitsgrad unter Fans besitzen.

Wir standen allerdings vor einem Problem: Um das Spiel überhaupt entwickeln zu können, benötigten wir auch ein »Next-Gen«-Startkapital – ein großer Unterschied zu den Live-Aid-Musikern, die einfach ins Studio gegangen sind und ihre Single aufgenommen haben. Und obwohl wir sicher waren, dass unser Spiel sein Budget dank der Star-Unterstützung schnell wieder einspielen würde, war es eine Herausforderung, das Startkapital aufzutreiben.

Probleme in der Praxis

Die Finanzierung unserer Wohltätigkeitsidee stellte sich zwar als schwierig heraus, Begeisterung für das Projekt gab es hingegen auf allen Seiten. Sowohl einzelne Designer als auch diverse Entwickler zeigten in den ersten Tagen Interesse für das Projekt. Manche von ihnen stiegen sogar gleich bei uns ein. Das einzige Problem war, dass die Entwickler trotz ihres Enthusiasmus nur wenig freie Zeit mitbrachten und die Deadlines ihrer kommerziellen Projekte unser Baby naturgemäß einschränkte.

Mit diesem Gedanken im Kopf und nach einige Gesprächsrunden mit unseren Beratern, entschieden wir uns dafür, das »eine große Spiel« zunächst einmal zurückzustellen und uns dem Markt mit einem pragmatischeren, besser umsetzbaren Plan zu nähern. Konkret bedeutete das: Wir fragten jeden Entwickler und Designer, ob er nicht jeweils ein kleines Spiel (im Prinzip die Mini-Games im Gegensatz zu dem einen großen Spiel) für uns entwickeln könne, das wir auf diversen Online-Kanälen verbreiten würden. Während dieser Zeit wuchs der Markt für kleine Casual Games enorm, daher schien uns der Schritt nur logisch.

Lösung Flash Game?

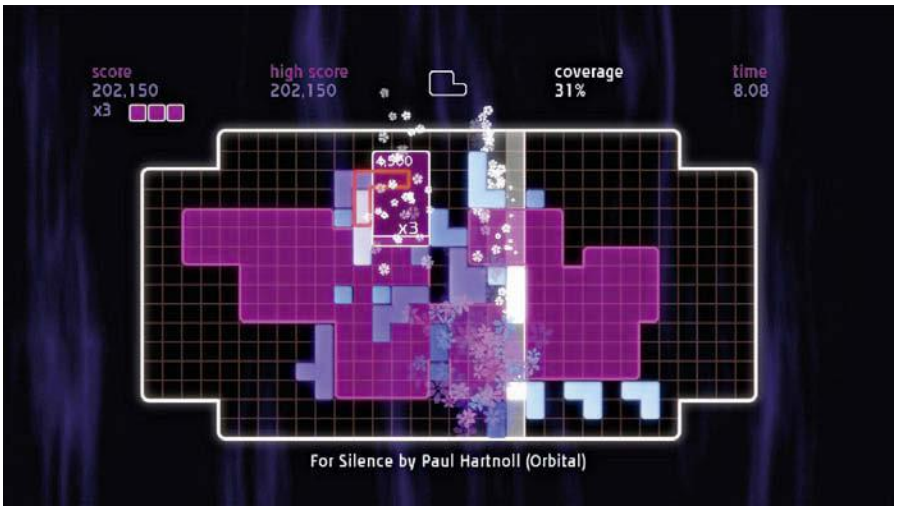
Anfangs konzentrierten wir uns auf Flash Games. Wir suchten bekannte Designer, die normalerweise an großen Next-Gen-Projekten arbeiteten und baten sie, sich auf ihre Kreativität zu verlassen und innovative Spiel-Mechanismen zu entwickeln. Auf diese Weise wollten wir jeweils ein süchtig machendes Gameplay schaffen, statt Spiele allein über die Power von Konsolen und High-End-PCs zu verkaufen. Wir waren überzeugt von dieser kreativen Herausforderung und dachten, dass Fans sehr interessiert an der Arbeit ihrer Star-Designer wären – insbesondere wenn diese an ihre eigenen Marken anknüpfen und etwa ein Spiel in demselben Genre entwickeln würden.

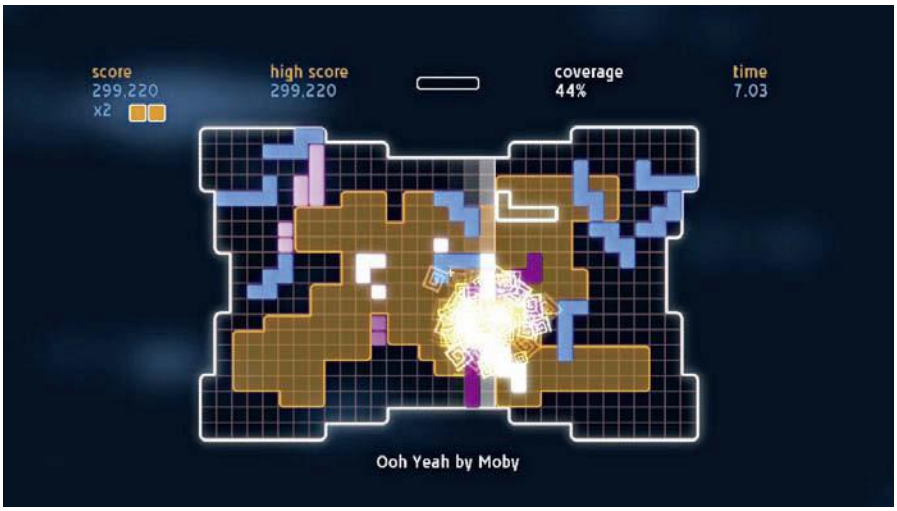
Unsere Idee war ein Spiele-Portal, auf dem die Leute unsere Flash Games umsonst spielen und über Werbung Gelder generieren würden, vergleichbar mit der Finanzierung unzähliger anderer Portale. Darüber hinaus wollten wir einen Download der Flash Games anbieten, um über den direkten Verkauf Geld einzunehmen.

Diese neue Strategie erwies sich als weitaus praktikabler. Gleichzeitig bedeutete die Plattform aber auch immens viel Arbeit für unser Team, das ausschließlich aus Freiwilligen bestand. Alle von uns hatten nebenbei zwar noch ihre normalen Jobs, wir waren allerdings nicht unter Zeitdruck und hatten auch keine Investoren im Nacken sitzen, die in einem bestimmten Zeitrahmen Resultate erwarteten. Daher bauten wir einfach weiter und weiter. Und gegen Ende 2007 nahm alles langsam aber sicher eine solide Form an.

Das erste Projekt

Die Entwickler kamen mit richtig coolen Ideen für kleine Spiele auf uns zu, der Bau des Web-Portals nahm Fahrt auf, Vertrieb, PR und Ge-





Das erste Projekt von OneBigGame ist das Musikspiel Chime. Die Besonderheit: Der Soundtrack passt sich dynamisch der Spielsituation an, hier laufen gerade Songs von Paul Hartnoll und Moby.

schäftspartner zeigten langsam Interesse an unserer Wohltätigkeitsinitiative. Aber am wichtigsten überhaupt: Die Entwicklung an unserem ersten Spiel startete offiziell.

Für den ersten Titel von OneBigGame ist kein einzelner Designer zuständig, sondern der erfahrene Entwickler Zoë Mode. Das Studio gehört zu Kuju Entertainment und hat sich mit Musik- und Partyspielen einen Namen gemacht. Nach großen Erfolgen wie »SingStar« und »Sing It« war es daher kein Wunder, dass sie für uns ebenfalls ein Musikspiel entwickeln wollten. Von Anfang an waren sie von dem Freiraum begeistert, den wir ihnen gaben. Herausgekommen ist ein wunderschönes Spiel, das jeglicher Beschreibung trotzt, obwohl »Chime« auf den ersten Blick wie ein »Lumines«-Klon ausschauen mag. Zwar ist das Spiel im Kern auch ein Block-Puzzlespiel in HD mit musikalischem Touch, dort hören die Gemeinsamkeiten allerdings auch schon wieder auf. Denn »Chime« verbindet das Blöckestapeln auf einzigartige Weise mit dem Musikmachen: Der Spieler sieht ein Raster vor sich, eine Beatline am oberen Bildschirmrand wummert im Tempo eines einfachen Musikstücks. Mit dem Gamepad wird ein Pentomino (eine Form bestehend aus fünf Quadraten) auf dem Raster platziert. Wandert die Beatline über das Muster, wird ein Soundeffekt getriggert, der die Hintergrundmusik verstärkt. Je mehr Steine auf dem Feld liegen, desto vielschichtiger wird die Soundkulisse. Das eigentliche Ziel ist jedoch, Quadrate oder Rechtecke zu bilden, die sich mit der Zeit auffüllen. Umschließt der Spieler diese Felder mit zusätzlichen Steinen, werden sie in das Raster eingestanzt und die Musik verstärkt sich noch weiter.

Gleichermaßen süchtig machend und hypnotisierend ist »Chime« eine einzigartige Synästhesie. Die Mischung aus fließendem Gameplay und Musik funktioniert auch deshalb so gut, weil alle Tracks von berühmten Musikern beigesteuert werden. Das Tolle ist eben die kollaborative Natur von OneBigGame. Was »Chime« angeht, wurden Zoë Mode und wir von Nimrod Productions unterstützt, einer Musikproduktionsfirma für Spiele aus Großbritannien. Sie besitzen gute Kontakte zu vielen bekannten Musikern, sodass die Spieler nun zu gestifteten Tracks von Moby und Orbital puzzeln können.

Die Website **www.onebiggame.org** bietet einen Überblick über das Charity-Projekt und in Kürze auch Informationen zu allen beteiligten Entwicklern.

Jeder hilft, jeder profitiert

Obwohl alles gut aussah, waren wir im Sommer 2008 immer noch nicht bereit zum Launch. Wir warteten auf die Fertigstellung mehrerer Spiele, anstelle von bloß einem oder zwei Titeln. Aber das freiwillige Engagement unserer Entwickler machte uns einen Strich durch die Rechnung. Die Spiele machten zwar Fortschritte, aber hin und wieder stoppte die Produktion, und Designer oder Entwickler mussten sich ihren kommerziellen Projekten zuwenden.

An diesem Punkt wandten wir uns an Game-Design-Studiengänge sowie an Indie-Entwickler auf der ganzen Welt. Die Idee war, einigen individuellen, für uns arbeitenden Game Designern Extra-Ressourcen in Form von Studenten oder Indie-Flash-Entwicklern zur Verfügung zu stellen und so die ganze Sache etwas zu beschleunigen. Den Studenten und Entwicklern machten wir auf diese Art ein verlockendes Angebot: Bist du interessiert daran, einer Designer-Legende unter die Arme zu greifen und ein kleines Spiel für wohltätige Zwecke zu entwickeln? Eine Win-win-Situation, wie wir fanden: Der Designer hatte mehr Ressourcen zur Verfügung, unsere Spiele würden schneller fertig und die Indie-Entwickler würden von der Erfahrung des Star-Designers lernen – darüber hinaus tauchen sie in den Credits auf.

Der Trick mit der Exklusivität

Neben den Vorzügen für Indie-Entwickler gibt es einen weiteren Aspekt bei OneBigGame, der zentral für den Erfolg des Projekts ist: die Exklusiv-Periode. Gleich zu Beginn machten wir allen unseren Helfern klar, dass wir niemals die Rechte an einem der Spiele besitzen würden. OneBigGame ist eine Nonprofit-Organisation, keine kommerzielle Instanz. Als Folge daraus besitzen wir keine einzige IP. Es gibt auch nur eine begrenzte Periode, in der wir Spiele veröffentlichen. Nach dieser Zeit (in der Regel zwölf Monate) wandert das Distributionsrecht zurück zum Entwickler, und wir entfernen das Spiel aus unseren Vertriebskanälen. Die Entwickler können dann mit dem Spiel machen, was sie wollen. Im Falle eines populären Titels können sie etwa eine Fortsetzung entwickeln oder eine Deluxe oder Enhanced Edition herausbringen, solange sie alle Verweise auf OneBigGame entfernen und dem Käufer signalisieren, dass der Erlös des entsprechenden Spiels nicht mehr für wohltätige Zwecke verwendet wird.

Who We Are
What We Do
Our Partners
What's New
How to Help
Social Networks
starlight
children's foundation
ONE BIG GAME
OneBigGame is a videogame industry charity initiative whose mission is to bring together famous game designers and major development studios to create a number of unique games, specifically developed for OneBigGame, with all the proceeds benefiting two children's charities: Starlight Children's Foundation and Save The Children. To learn more, visit www.onebiggame.org.
eNEWS
SURVEY
HOW TO HELP
FIND MY LOCAL OFFICE:

Die Erlöse der Spiele gehen an die **Starlight Children's Foundation** ...

... sowie an die Kinderhilfsorganisation **Save the Children**.

Indem wir dies anbieten, können Designer und Entwickler OneBigGame im Prinzip als Launch-Plattform für eine neue IP nutzen, wenn auch mit einer gewissen Verzögerung. Zusätzlich macht dieser Mechanismus es für die beteiligten Indie-Entwickler noch interessanter: Die Top-Designer können sich dazu entschließen, nach der OneBigGame-Version eines Spiels für die kommerzielle Variante weiter mit dem Indie-Entwickler zusammenzuarbeiten.

Der Distributionskanal

Ein weiterer großer Faktor, der sich als Vorteil im eben beschriebenen Prozess auswirken wird, ist unser Zusammenschluss mit einem der weltweit größten Distributoren: Mit mehr als 100 Millionen Seitenaufrufen wird dieses von uns in Kürze bekannt gegebene Web-Portal-Unternehmen unsere Spiele auf seinen Websites anbieten und alle Erlöse aus seiner Werbung an uns ausschütten. Und jetzt rechnen Sie noch unsere zwei ehrenamtlich tätigen PR-Unternehmen Lincoln Beasley und FortySeven Communications als starke Partner hinzu. Zusammen haben diese Kontakt zu den wichtigsten Journalisten in der Branche und damit Zugriff auf die einflussreichsten Medien. Weiter werden wir von Atomic PR unterstützt, einem Spezialisten für soziale Netzwerke, der unsere Initiative und damit unsere Spiele auf den Twitters, Facebooks und MySpaces dieser Welt bekannt macht. Kurz gesagt: Alle Beteiligten können sich darauf verlassen, dass ihre Spiele die Aufmerksamkeit erhalten, die sie verdienen. Und das macht OneBigGame zu einer idealen Launch-Plattform für neue, innovative IPs.

Von Flash zur Konsole

Dank der Hingabe unserer Designer und der Unterstützung von Studenten und Indie-Entwicklern näherten wir uns 2009 der finalen Phase von OneBigGame. Dann jedoch, kurz vor dem Launch unseres Portals und der Veröffentlichung des ersten Spiels, passierte etwas Überraschendes: Einige Entwickler, die zunächst für ein Flash Game eingeplant waren, wollten stattdessen gern ein Download-Spiel für iPhone, XBLA oder PSN designen – entweder weil ihnen die Entwicklungsumgebung vertrauter war als bei Flash oder weil sie einfach mit dem iPhone herumexperimentieren wollten. Natürlich hatten wir nichts dagegen, es bedeutete nur weitere Anpassungen unseres Plans: Wir würden zu einem Publisher werden, der alle Plattformen bedient, anstatt nur unser eigenes Portal und Flash Games im Blick zu haben. Und wir mussten natürlich die Zustimmung der »großen Drei« einholen, unsere Spiele auf ihren Plattformen veröffentlichen zu dürfen, bzw. sie bitten, diese selbst zu vertreiben. Das bedeutete zwar viel Arbeit, gleichzeitig aber auch große Chancen, da diese Premium-Online-Versionen auf den Konsolen mehr als nur Einnahmen aus Werbung versprachen. Also kontaktierten wir die drei großen Plattformbetreiber sowie einige andere App Stores oder PC-Spiele-Kanäle und befragten sie bezüglich ihres Interesses nach den High-End-Versionen unserer Spiele. In den meisten Fällen fielen die Reaktionen äußerst positiv aus, sodass viele Spiele nun statt der Flash-Umgebung eine der Konsolen als führende Plattform nutzen. Das sind natürlich tolle Neuigkeiten für uns, und es wird sehr dabei helfen, in Zukunft noch mehr Gelder für unsere Partner »Save the Children« und »Starlight Children's Foundation« aufzubringen.

Dieser neue Kurs bedeutete dann schließlich auch ein Redesign für unser Web-Portal und einen sehr viel größeren Aufwand für das Polishing unserer Spiele inklusive Support von einigen High-End-QA-Dienstleistern. Nach drei langen Jahren des Wartens machten diese paar Monate Mehrarbeit für uns dann aber auch keinen allzu großen Unterschied mehr.

Fast fertig

Während wir uns nun langsam dem Ende des Jahres 2009 nähern, befinden sich etwa 20 Spiele in Produktion – die Bandbreite unserer Titel erstreckt sich von Flash über den PC-Download bis hin zu XBLA, PSN und iPhone. Manche unserer Spiele werden von einzelnen berühmten Designern entwickelt, hinter anderen stehen wiederum bekannte, professionelle Studios, und wieder andere stammen vollständig von Indie-Entwicklern. Den Vertrieb sichern unsere eigene Spieleplattform, der Deal mit dem Web-Portal, die Unterstützung der wichtigsten Konsolenbetreiber und die Verlässlichkeit einer soliden PR-Kampagne für jedes Spiel.

Und die beste Neuigkeit ist: Die Spiele scheinen richtig gut zu werden. Unsere Designer und Entwickler haben Fantastisches geleistet, wenn man die engen Grenzen bedenkt, in denen sie sich bewegen mussten. Und diese innovativen Ideen sind es dann auch, weswegen es unsere Spiele mit jedem kommerziellen Unterhaltungsprodukt da draußen aufnehmen können. Klar, wir sind natürlich voreingenommen, aber bedenkt man die ersten Reaktionen von Außenstehenden auf unsere Spiele, müssen sie am Ende einfach etwas Besonderes werden.

Und jetzt? Deutschland!

Jetzt, da wir uns endlich dem Launch nähern, ist unser Blick schon fest auf die Zeit nach dem ersten Schwung Spiele gerichtet. Wir brauchen mehr Inhalte auf unserer Plattform, um uns sicher in 2010 und 2011 hineinzubringen. Trotz der Partner in den Vereinigten Staaten, Japan, Großbritannien, Skandinavien, Frankreich und den Beneluxstaaten kommen wir nicht umhin, eine der weltweit größten und wichtigsten Videospiel-Nationen zu vermissen: Deutschland.

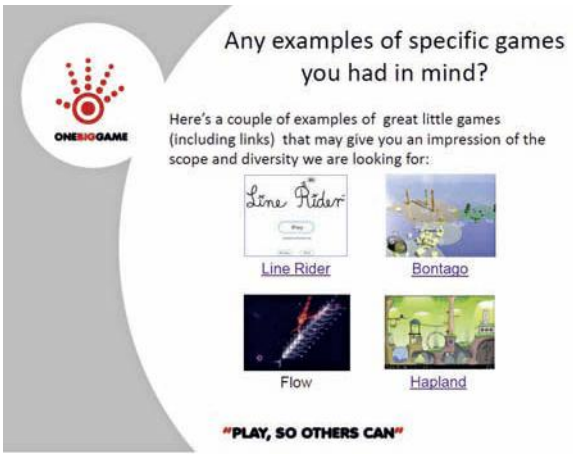
Nicht nur ist Deutschland der zweit- oder drittgrößte Spielemarkt der Welt, er ist gleichermaßen die treibende Kraft für »Free to play«-Browsergames. Und wäre das allein nicht schon erfreulich genug, beherbergt Deutschland eine ganze Riege talentierter Entwickler, die seit Jahrzehnten für Hit-Franchises sorgen. Unnötig zu sagen, dass wir derart starke Mitstreiter aus Deutschland mit offenen Armen in unserem Projekt willkommen heißen würden, schließlich soll OneBigGame am Ende tatsächlich eine weltumspannende Initiative sein.

Helfen kann man uns auf ganz unterschiedliche Weise: Entwickler können uns ein originelles Spiel liefern, wahlweise in Flash oder als kleinen Download auf einer der führenden Plattformen (Windows oder Mac, WiiWare, PSN, XBLA oder iPhone). Wir kümmern uns um die gesamte Lokalisation, die Qualitätssicherung, PR und das Marketing. Gemeinsam entscheiden wir uns dann für eine Preispolitik, entwickeln eine Kampagne und einigen uns auf den Rahmen der Exklusiv-Periode. Nach dieser Zeit verabschieden wir uns aber nicht einfach von unserem Partner, sondern unterstützen sein Spiel weiterhin als kleinen Dank für die Hilfe. Man kann uns aber auch anders unter die Arme greifen. Etwa, indem sich Studenten und Indie-Teams zur Verfügung stellen oder spezialisierte Profis ihre Fachkompetenz beisteuern. So springen dann KI-Experten, Physik-Programmierer oder Flash-Gurus ein, wenn sie für ein bestimmtes Spiel gebraucht werden. Und natürlich sind uns auch Spezialisten im Vertrieb und Marketing herzlich willkommen.

Kurz gesagt: Jede Art von Hilfe wird gern gesehen, und je mehr Entwickler OneBigGame unterstützen, desto eher wird dieses Projekt von Erfolg gekrönt sein. Und mehr Erfolg bedeutet wiederum mehr tolle Spiele von einigen der kreativsten Köpfe der Industrie. Ich finde, das sind tolle Nachrichten für uns alle, die tief im Herzen ja auch nur Gamer sind und eine innovative Spielmechanik zu schätzen wissen. Wenn Sie also interessiert sind, sich unserer Bewegung anzuschließen, zögern Sie nicht, mich anzuschreiben. Und nicht vergessen: Schauen Sie gegen Ende des Jahres auf unserer Website vorbei, wenn wir als Spieleportal an den Markt gehen.

Martin de Ronde

Das große Musik-Vorbild **Live Aid** lockte in den 80er-Jahren Millionen Menschen in die Stadien und bekämpfte so die große Hungersnot in Äthiopien.


Any examples of specific games you had in mind?
Here's a couple of examples of great little games (including links) that may give you an impression of the scope and diversity we are looking for:
Line Rider
Bontago
Flow
Hapland
"PLAY, SO OTHERS CAN"

Unter der Rubrik »FAQ« auf der Website finden sich einige Beispiele für **innovative Spiele**, wie OneBigGame sie sich vorstellt.

Vorträge im Fokus
Vier Themen des BGF 2009

Am 6. und 7. November trifft sich die Branche zum zweiten Browsergames Forum im Frankfurter Cinestar – unterstützt vom Making Games Magazin. Vier Sprecher haben uns vorab ein Interview gegeben.

Sven Liebich

ist Head of Art bei Bigpoint.

Was ist das Thema deines Vortrags?
3D-Grafik in Browserspielen – Risiken und Nebenwirkungen.

Warum ist dieses Thema für dich persönlich so wichtig?
Mit der rasanten Verbreitung und der damit einhergehenden Entwicklung von Technologien wachsen die grafischen Möglichkeiten, aber auch die finanziellen Risiken für Entwickler und Publisher.

Was sollen die Zuhörer deines Vortrags mit nach Hause nehmen?
Gilt der Spruch »Schuster, bleib bei deinen Leisten« auch für Browserspiele und vor allem auch für mich als Entwickler? Oder ist die Zeit reif für den nächsten Evolutionssprung?

Was sind deine persönlichen Erwartungen an das diesjährige Browsergames Forum?
Ich freue mich auf spannende Vorträge und anregende Diskussionen rund ums Thema Grafik und Zielgruppenanalyse.

Was sind deiner Meinung nach die derzeit spannendsten Trends und Entwicklungen im Browsergames-Segment?
Zum einen die Optimierung der ersten fünf Spielminuten, zum anderen faszinieren mich neue Ideen zur Kundengewinnung, Kundenbindung sowie Mittel und Wege, wie man die Abwanderungsquote so klein wie möglich halten kann.

Florian Bohn

ist CEO von Travian Games.

Was ist das Thema deines Vortrags?
Spieler-Support und Community Management – Stolpersteine und Herausforderungen.

Warum ist dieses Thema für dich persönlich so wichtig?
Wir haben uns das Ziel gesetzt, unseren Spielern einen besonders guten Service zu bieten. Entsprechend zentral ist das Thema für Travian Games und für mich persönlich. In einem zunehmend engeren Markt kann der Service zu einem noch wichtigeren Unterscheidungsmerkmal werden. Besonders für kleinere Entwickler ist das Thema bei der Veröffentlichung eines Titels häufig eine große Herausforderung.

Was sollen die Zuhörer deines Vortrags mit nach Hause nehmen?
Viele beschäftigen sich bereits intensiv mit den Fragen des Spieler-Supports und des Community Managements. Dabei gibt es keine Musterlösungen oder den einzig richtigen Weg. Für zufriedene Spieler muss man hart arbeiten und steht stetig vor neuen Herausforderungen. Ich würde mich freuen, wenn ich ein paar Anregungen geben kann.

Was sind deine persönlichen Erwartungen an das diesjährige Browsergames Forum?
Da es mein erstes BGF ist, bin ich besonders auf das familiäre Flair gespannt, dem die Veranstaltung wohl ihren Mythos zu verdanken hat.

Was sind deiner Meinung nach die derzeit spannendsten Trends und Entwicklungen im Browsergames-Segment?
Genau diese Frage beantwortet ein Besuch auf dem BGF wohl am besten.

Hendrik Klindworth

ist Mitbegründer und Geschäftsführer der InnoGames GmbH.

Was ist das Thema deines Vortrags?
Balancing bei Browsergames.

Warum ist dieses Thema für dich persönlich so wichtig?
Eine gute und ausgewogene Spielbalance ist aus meiner Sicht der zentrale Punkt, um eine hohe Langzeitmotivation bei den Mitspielern zu erreichen. Grafik und Handling sind für jedes Spiel natürlich auch wichtige Faktoren – gerade wenn es darum geht, einen Spieler in der Anfangsphase für ein Spiel zu begeistern. Sie halten einen Spieler allerdings nicht über Monate oder gar Jahre in einem Spiel. Wenn man dies aber zum Ziel hat, muss man viel Energie und Engagement in das Balancing investieren.

Was sollen die Zuhörer deines Vortrags mit nach Hause nehmen?
Am liebsten wäre es mir, wenn sich nach meinem Vortrag eine lebhafte Diskussion ergibt. Entwickler, die vielleicht am Anfang ihrer Karriere stehen, würde ich gern motivieren, von Beginn an ein starkes Augenmerk auf das Balancing zu legen.

Was sind deine persönlichen Erwartungen an das diesjährige Browsergames Forum?
Die Browsergames-Szene in Deutschland ist zwar mächtig angewachsen, aber dennoch so etwas wie eine große Familie. Insofern freue ich mich einfach darauf, viele bekannte Gesichter zu sehen und mit ihnen über die aktuellen Entwicklungen auf dem Markt zu diskutieren. In der Regel ist es doch so: Wenn zwei sich über ein Thema austauschen, sind am Ende beide klüger.

Was sind deiner Meinung nach die derzeit spannendsten Trends und Entwicklungen im Browsergames-Segment?
Sehr spannend finde ich, wie die Vernetzung von Browsergames und Social Networks immer weiter voranschreitet. Hier entwickelt sich kein Mit- oder Nebeneinander, sondern langfristig wohl eher ein Ineinander.

Simon Seefeldt

ist Head of Business Development bei Jagex.

Was ist das Thema deines Vortrags?
Subscriptions vs. MTX – Why Subscriptions aren't dead!

Warum ist dieses Thema für dich persönlich so wichtig?
Unser Geschäftsmodell ist derzeit komplett auf Subscription ausgelegt.

Was sollen die Zuhörer deines Vortrags mit nach Hause nehmen?
Ein besseres Verständnis der zwei unterschiedlichen Business-Modelle.

Was sind deine persönlichen Erwartungen an das diesjährige Browsergames Forum?
Ich hoffe auf interessante Vorträge und Diskussionen und freue mich auf die Präsenz der meisten Key Player im Browsergames-Segment.

Was sind deiner Meinung nach die derzeit spannendsten Trends und Entwicklungen im Browsergames-Segment?
Die Fortschritte der Browsergames-Technologie, die immer komplexere Games unterstützt.

Entwicklungsverträge, Teil 18
Leistungsstörungen, Vertragsbeendigung

Was passiert, wenn es zwischen Entwickler und Publisher zum Bruch kommt? Wir klären, welche Interessen in den Projektphasen aufeinanderprallen und welche Auswirkungen dies auf die Fertigstellung des Produkts hat.

Claas Oehler

ist Geschäftsführer Recht beim G.A.M.E.-Bundesverband der Entwickler von Computerspielen.

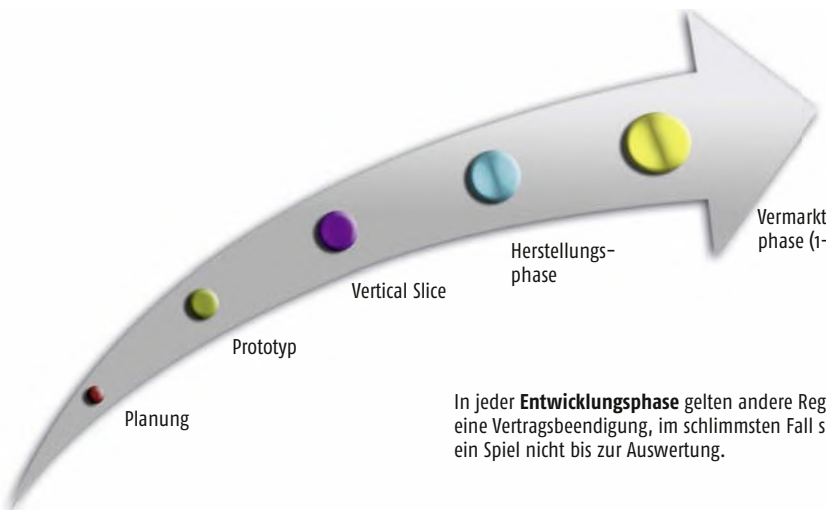
Neben seiner Tätigkeit für G.A.M.E. arbeitet Claas Oehler auch als Rechtsanwalt für die Kanzlei IHDE & Partner in Berlin und berät Entwickler, Produzenten sowie Verlage.
claas.oehler@onlinelaw.de

Während einer längeren Produktion tritt viel Unerwartetes ein: Der Markt verändert sich, die Erwartungen von Entwickler und Publisher über die Qualität des Spiels divergieren, technische Probleme gefährden den Erfolg, es fehlt Personal oder der einen Partei geht schlicht das Geld aus.

Wenn die Egos von Lead Designer und Producer aufeinanderprallen, dann kann nur noch ein Austausch der Personen helfen. Das Leben schreibt aber so vielfältige Geschichten, dass nicht alle Eventualitäten einer Produktion vertraglich vorhergesehen werden können. Manchmal bleibt daher nur die Ultima Ratio: die Zusammenarbeit zu beenden. Ebenso verhält es sich in der Auswertungsphase. Gelegentlich kann man einem Publisher vorwerfen, dass er seine Vermarktungsversprechen nicht einhält oder unsauber abrechnet. Auf der anderen Seite können auch äußere Umstände die Vermarktung behindern und die Auswertung erschweren, beispielsweise wenn neue Jugendschutzbestimmungen oder Verbraucherschutzvorschriften verabschiedet werden. Führt eine Vertragspartei ihre Pflichten nachhaltig nicht durch, droht ihr als Sanktion die Kündigung.

Kündigungsarten

Das Gesetz sieht in dieser Situation verschiedene Möglichkeiten vor: Dazu zählen beispielsweise die ordentliche Kündigung, der Rücktritt sowie die Kündigung aus wichtigem Grund. Die Rechtsfolgen, die das Gesetz an solche Beendigungen knüpft, gleichen die Interessen der Beteiligten allerdings nicht immer gerecht aus. Gerade bei Entwicklungsverträgen, die unterschiedliche Stufen des Leistungsaustausches erfassen, passen die Rechtsfolgen des Gesetzes selten. Es entstehen in einigen Fällen sogar »Lose-lose-Situationen«, in denen beide Parteien den Kürzeren ziehen. Die Rückabwicklung bedarf einer abgleichenden vertraglichen Ausgestaltung, wobei die Möglichkeiten in Standardverträgen dabei sehr begrenzt sind. Um diese Probleme zu beschreiben und zu regeln, teilen wir die verschiedenen Stadien der Herstellung und Verwaltung auf, wobei der Produktwert in der Regel bis zum Beginn der Verwertung zunimmt.



In jeder **Entwicklungsphase** gelten andere Regeln für eine Vertragsbeendigung, im schlimmsten Fall schafft es ein Spiel nicht bis zur Auswertung.

Entwicklungs-, Projektierungs- oder Prototypenphase

In dieser frühen Phase befindet sich der Leistungsaustausch in der Regel noch am Anfang. Entweder hat der Entwickler ein Konzept mit eingebracht oder er hat es während der Projektierungsphase entwickelt. Je nach Umfang der Produktion werden die bislang noch eher geringen Kosten durch den Publisher bezahlt. Im Normalfall wird nur ein kleiner Teil des Gesamtproduktionsbudgets verbraucht, sodass sich die Interessenlage in der Regel wie folgt darstellt: Der Entwickler hat ein Konzept mit möglicherweise großem Potenzial ausgearbeitet. Er stellt es zur Verfügung, um an der Auswertung zu partizipieren. Hätte er es ausschließlich als Konzept für Dritte entwickelt, hätte er einen höheren Preis verlangt. Bringt er das Konzept in die Produktion ein, möchte er bei Vertragsauflösung das Konzept entweder selbst zu einem fertigen Spiel weiterentwickeln oder das enthaltene Potenzial vergütet bekommen. Der Publisher hingegen hat etwas bezahlt, für diese Zahlung möchte er natürlich einen Gegenwert erhalten. Diese antagonistischen Interessen gilt es bei der vorzeitigen Trennung aufzulösen. Wenn die Prototypenphase als Teil eines einheitlichen Werkvertrags ausgestaltet ist, so kann der Entwickler grundsätzlich darauf vertrauen, dass das Projekt auch zu Ende geführt

wird und er den in der Entwicklervergütung enthaltenen Gewinn aus dieser Phase zugeteilt bekommt. Hier gilt im Wesentlichen das bereits in einer früheren Folge zu § 649 BGB Gesagte:»Er ist berechtigt, die vereinbarte Vergütung zu verlangen; [...] muss sich jedoch dasjenige anrechnen lassen, was er infolge der Aufhebung des Vertrags an Aufwendungen erspart oder durch anderweitige Verwendung seiner Arbeitskraft erwirbt oder zu erwerben böswillig unterlässt.« Dabei vermutet das Gesetz, dass fünf Prozent der Vergütung für die noch ausstehenden Produktionsteile zu zahlen sind. Diese Vermutung lässt sich zwar durch eine Berechnung widerlegen, die Details der Kalkulation müssen dann allerdings veröffentlicht werden.

Mit dieser Vergütung ist aber nur der werkvertragliche Teil erfasst, der für Verträge ohne die Beteiligung eines Entwicklers an der Auswertung gilt. Ob in einem solch frühen Stadium auch schon die möglichen Auswertungserlöse berücksichtigt werden sollen, hängt im Wesentlichen von den Umständen und dem Konzept ab. So verhält es sich zum Beispiel, wenn beide gemeinsam ein Konzept entwickelt haben.

Herstellungsphase

In der Herstellungsphase sind die Investitionen des Publishers meist fest mit der Produktion verzahnt. Für ihn wird ein Rücktritt immer kostspieliger, während der Entwickler weniger stark von einer Einstellung des Projektes betroffen ist, zumal er oft schon während der Entwicklung Gewinne erwirtschaftet. Im Falle einer Einstellung des Projekts verliert der Publisher all sein bisher investiertes Geld. Der Entwickler muss auf der anderen Seite womöglich Arbeitnehmer weiterbeschäftigen, die er nur für das Projekt eingestellt und verplant hatte. Er muss also gegebenenfalls Ersatzprojekte finden oder sie nach dem Ablauf der Kündigungsfrist entlassen. Wurde seine Gewinnerwartung primär aus der Herstellung befriedigt, so entgeht ihm ein Gewinn, der auf die zukünftigen Milestones verteilt war. Wenn der Entwickler auch an der Auswertung partizipiert und vielleicht sogar Eigenanteile eingebracht hat, die er durch die Auswertungsbeteiligung wieder einzunehmen hofft, sind auch seine Investitionen verloren. Er wird in der Regel also daran interessiert sein, das Spiel zur Marktreife zu bringen und auszuwerten. Da der (mit-)finanzierende Publisher das bisher gezahlte Herstellungsbudget nicht verschenken will, wird er die Rechte nur gegen Rückzahlung des investierten Kapitals (gegebenenfalls mit Verzinsung) oder einer Beteiligung an der Auswertung freigeben. Darauf ist der Entwickler in der Regel angewiesen, da die Rechte üblicherweise mit den einzelnen Milestones auf den Publisher übergehen, wenn auch oft nur in Teilen. In bestimmten Konstellationen kann es die Interessenlage auch erfordern, dass die Rechte unabhängig davon auf den Entwickler zurückfallen, zum Beispiel wenn ein Schaden entstanden ist, der höher als das vom Publisher investierte Kapital ist. Wird das Produkt überhaupt nicht ausgewertet, entsteht die oben erwähnte »Lose-lose-Situation«.

Auswertungsphase

In der Auswertungsphase ist in der Regel der Publisher maßgeblich verpflichtet. Der Entwickler wird fast nur noch Updates und Pflege übernehmen, während der Publisher aktiv das Produkt vertreibt und die hoffentlich entstehenden Gewinne verteilt. In solchen Phasen sind die Leistungsstörungen und Interessen anders verteilt als in der Herstellungsphase. Wenn der Publisher beispielsweise schlecht auswertet oder falsch abrechnet, braucht der Entwickler ein Druckmittel, im Ernstfall eine Rückabwicklung des Vertrages. Auf der anderen Seite hat der Publisher erhebliche Investitionen getätigt, die eventuell mit den Renditeerwartungen seiner Aktionäre oder denen anders finanzierender Anleger einhergehen müssen.

Computerspiele sind leider relativ kurzlebige Produkte. Die Chancen auf ein Einspielergebnis in Höhe der Entwicklungskosten sinken bei den meisten Produkten mit jedem Tag seit der Erstveröffentlichung bzw. der Fertigstellung. Denn es ändern sich die technischen und optischen Eigenschaften der Konkurrenzprodukte. Das Marketing ist größtenteils zielgenau auf die ersten drei Monate ausgerichtet, danach fallen die Verkäufe regelmäßig stark ab. Eine Kündigung, die mit einem Rechterückfall einhergeht, stört die Auswertung erheblich, denn es müssen in diesem Fall neue Produkte gepresst werden und gegebenenfalls neue Freigaben bei den Konsolenherstellern erfolgen. Der Schaden kann durchaus erheblich sein. Gleichzeitig muss die Vertragssituation eine Sanktion vorsehen, damit über den Abrechnungs- und Vermarktungspflichten das Damoklesschwert des Verlustes schwebt.

Wenn Regeln über den Rechterückfall geschrieben werden, so ist zu bedenken, dass die Einzelteile eines Computerspiels nicht aus nur einer Hand geliefert werden. Üblicherweise werden die Lokalisierungen, Sprachaufnahmen, manchmal auch die Musik- und Motion-Capturing-Aufnahmen oder einzelne Teile des Computerspiels durch den Publisher geliefert, während die übrigen Teile vom Entwickler stammen. Eine Vertragskündigung, bei der die Rechte an die jeweilige Partei zurückfallen, macht die Auswertung des Computerspiels unmöglich. Die entsprechenden Teile müssen ersetzt werden. Sinnvoll ist deswegen eine vertragliche Regelung, die einer Partei sämtliche Rechte zuweist und die andere gegebenenfalls an den Erlösen beteiligt. Hierfür bieten sich verschiedene Modelle an. Diese Modelle unterscheiden sich danach, welche Seite die Vertragsrückabwicklung verursacht bzw. zu vertreten hat.

Claas Oehler

Typische Vertragsinhalte

0. Vorspann und wesentliche Punkte, Zusammenfassung

1. Definitionen

2. Entwicklungsprozess
Leistungsabschnitte, Termine
Kommunikation (Ansprechpartner, Weisungen)
Teil-/Abnahmeverfahren
Inhalte, die vom Publisher geliefert werden
Einbau
Development Kits
Änderungen
Verzögerungen und deren Folgen
Bindung des Kernteams
Lokalisierungen, Service, Support
(Versicherungen)

3. Rechteübertragung und Verwertung
Rechteeinräumung (Lizenz), Gebiet, Dauer
Exklusivität (in US-Verträgen oft »work made for hire« und Urheberschaft)
Verwertungsrechte, Titelrechte, Marken- und Kennzeichen
Charaktere, Dokumentation, Quellcode
Ausnahmen und Vorbehalte von der Rechteeinräumung
Credits, Anbringen von Logos und Urheberbenennungen
Optionen, Nachfolger
Wettbewerbsverbote

4. Finanzierung, Vergütung, Beteiligung
Vorauszahlungen
Abrechnungsschlüssel
Royalties, Verrechnungen
Beteiligung an anderen Produkten
Konzernverrechnung, Kompensationsgeschäfte
Zahlungswege, Bankverbindung
Retourreserven
(Doppel-)Besteuerung

5. Pflichten der Parteien
Entwicklerpflichten
Publisherpflichten
Vertrieb, Marketing

6. Laufzeit und Kündigung
»Kündbarkeit«:
- vor Fertigstellung, nach Fertigstellung
- wegen nicht erreichter Konsolenhersteller-anforderung

Pflichtverletzungen, Abmahnungen, Fristen, Folgen, Ausgleichszahlungen, Rechterückfall, Insolvenzfall und Sicherungsrechte
Fortwirkung von Klauseln

7. Garantien, Freistellungen, Haftungsbeschränkungen
Fehlerklassen, Rechteeinräumung
Mitverschulden
Mitwirkung an Prozessen

8. Vertraulichkeit
Vertrauliche Informationen, Ausnahmen
Verpflichtung der Mitarbeiter

9. Allgemeine Klauseln
Abtretbarkeit, Aufrechnung
Erklärungen, Zugang
Schiedsklausel
Anwendbares Recht, Gerichtsstand
Vollständige salvatorische Klausel

Vertragsmuster: http://contracts.corporate.findlaw.com/type/operations/ip.html

Emotionen in Games
Noch einmal mit Gefühl

Ein nächster großer Entwicklungsschritt für Computerspiele liegt in der Erzeugung von Emotionen, behauptet Björn Blankenheim. Die Methode dazu, von Filmen zu lernen, ist gut – aber das allein genügt nicht.

David Cages Vorzeigeprodukt **Heavy Rain** gilt als großer Hoffnungsträger für den Einsatz von Emotionen in Spielen.

Björn Blankenheim

ist Wissenschaftlicher Mitarbeiter an der Bergischen Universität Wuppertal.

Björn Blankenheim führt am Institut für Kunst und Mediendesign Seminare zur Geschichte und Systematik der Gestaltungsgrundlagen und der Mediengestaltung durch. Er promoviert im Lehr- und Forschungsgebiet »Gestaltungstechnik und Kunstgeschichte« zu den Grundlagen des Game Designs.

WWW
makinggames.de

- Wo ist der Witz? Eine Analyse, warum es so wenige humorvolle Spiele gibt.

Die Weiterentwicklung elektronischer Spiele hängt entscheidend von der Einbindung von Emotionen ab. Das Ziel ist klar: den Citizen Kane (1941) oder vielleicht auch das Bambi (1942) des elektronischen Spiels vorzulegen.

David Cage, der Autor und Director von Fahrenheit (2005) und Heavy Rain (2010), präsentierte in seiner Keynote »Writing Interactive Narrative for a Mature Audience« auf der GDC Europe 2009 zu diesem Zweck sein »Magic Recipe to Create Mature Games«. Doch wer, ähnlich einem Kochrezept, eine klare Anleitung mit stets reproduzierbarem Ergebnis erwartet hatte, war von Aussagen im Stile von »Sei ein guter Koch! Sei kreativ! Lass dir nicht reinreden! Mach etwas, das Bedeutung hat!« doch etwas enttäuscht.

Dabei ist es gar nicht nötig, sich auf derartige rein normative Vorschriften zu verlassen. Es gibt nämlich durchaus Rezepte, die in der Lage sind, Emotionen in Spielen wirksam Raum zu geben. So gehört es zur Professionalisierung einer jeden Kunstform, Regel- und Anleitungswissen hervorzubringen, statt einzig an die Kreativität des Künstlers zu appellieren. Und tatsächlich kann man auch vieles von anderen Medien wie dem Film oder auch der Malerei lernen. Dieses Wissen mag häufig nur implizit vorhanden sein oder noch seiner Entdeckung harren, doch es ist da (Heinen 2008) – davon zeugen nicht zuletzt diverse Artikel im Making Games Magazin.

Natürlich kann im Rahmen eines solchen Artikels nur eine erste Übersicht skizziert werden. Dazu ist es aber zunächst einmal wichtig, die Frage zu beantworten, was man unter Emotionen eigentlich versteht.

Die Beutung von Emotionen

Den Emotionen hat man in den letzten Jahren auch in der Forschung verstärkt Aufmerksamkeit gewidmet. Vorläufiger Höhepunkt war dabei die Schrift »Descartes' Irrtum« von Antonio Damasio (1999). Ihre Kernaussage ist, dass es fundamentale Wechselwirkungen zwischen Körper und Geist bzw. physiologischen Zuständen und kognitiver Verarbeitung von Umweltreizen gibt. René Descartes (1596–1650) hatte den Geist des Menschen – ganz metaphysisch – als vollkommen losgelöst von dessen materieller Existenz gedeutet. Die neue Sichtweise begreift die Emotionen des Menschen als zentrale Kategorien der Weltwahrnehmung, der Entscheidungsfindung und des Bewusstseins. Eine Erkenntnis, der sich beispielsweise das Marketing unter dem Stichwort der »Aktivierenden Prozesse« schon vor längerer Zeit geöffnet hat (Kroeber-Riel 2009).

Damit aus angeborenen, undefinierten Gefühlsregungen auch Emotionen werden können, muss allerdings zugleich das Bewusstsein an ihnen beteiligt sein; muss sie zu Bewusstsein bringen. Und nicht immer treten Gefühlsre-

gungen in klar differenzierbarer Reinform auf, sondern mischen sich vielmehr zu einer komplexen Form der Selbstwahrnehmung, und damit der Wahrnehmung und Bewertung der Umwelt.

Was ist Emotion?

Schon der Begriff »Emotion« ist allerdings nicht unproblematisch, setzt er sich doch zusammen aus »ex« (lat.) für »aus/heraus« sowie »movere« (lat.) für »bewegen«. Dieses Verständnis unterstellt, dass eine Gefühlsregung immer im Individuum erzeugt wird und sich aus diesem heraus bewegt oder auch herausbricht. Angesprochen werden diese Regungen allein durch Objekte, Verhaltensweisen und Situationen als Bedeutungsträger und Symbole, die wiederum von subjektiver Interpretation abhängig sind. Das ist eine Vorstellung, die verbunden mit einem stark romantisierenden Weltbild ist, in dem Emotionen etwas Authentisches, Natürliches und Subjektdefinierendes innewohnt.

Geschieden werden kann dieses Verständnis der Emotion vom sogenannten »Affekt«, ausgehend vom lateinischen Wort »afficere« für »jemandem etwas antun, auf jemanden einwirken, jemanden in einen Zustand versetzen«. Der affektauslösende Reiz kommt dabei immer von außen und führt bei Rezipienten unweigerlich zu physiologischen Reaktionen. Affekte sind also etwas, was sich herstellen lässt, und zwar allein durch die äußerlichen Reize, die dem Individuum präsentiert werden.

Vor diesem Hintergrund ist es keine Selbstverständlichkeit, dass die Entwickler elektronischer Spiele heute ein sehr differenziertes Bild von Methoden haben, um den Spieler in Horror- und Gruselszenarien in einen körperlichen Stresszustand zu versetzen und so Affekte zu erzeugen. Zudem sind Angst, mitreißende Erregung und anschließende Euphorie zentrale Elemente vieler Multiplayer-Spiele, daher wäre es falsch, ihre Bedeutung zu unterschätzen. Das Erfahren gemeinsamer Stresserlebnisse ist nicht nur ein großer Spaß, sondern schweißt zudem Clans oder Nationalmannschaften sozial verbindlich und effizient agierend zusammen (Mühlmann 2005: 74f.).

Dabei wäre es selbstverständlich falsch, sich bei der Betrachtung des für Spielentwicklung relevanten Spektrums menschlicher Gefühlsregungen auf den kleinen Ausschnitt der erschreckenden oder euphorisierenden Emotionalisierung zu beschränken. Daran haben sowohl David Cage als auch Kellee Santiago, Producerin von Flower (2009), appelliert.

Eine Ausbildungslücke

Obwohl die vielfältigen Gefühlsregungen eine so wichtige Bedeutung für das menschliche Dasein haben, spielen sie in der von formalen Prinzipien (wie Gestaltwahrnehmung, Symmetrie und Proportion) oder formalisierten Kreativitätstrainings geleiteten Grundlagenschulung innerhalb der Designerausbildung keine Rolle. Selbst produktorientierte Gestaltungsregeln wie das bekannte Motto »form follows function« blenden Emotionen kategorisch aus. Jenseits dieser Gestaltungsregeln, die freilich ihren eigenen Wert für die Leitung von Gestaltern besitzen, gibt es jedoch spezifisches Wissen über das Erzeugen von Gefühlsregungen, das beispielsweise in der Filmproduktion eingesetzt wird und zu einem Regel- und Anleitungskanon verdichtet werden kann. Solche Gestaltungstechniken können auch eingesetzt werden, um die Emotionalität oder auch Affekthaltigkeit elektronischer Spiele zu steigern. Im Folgenden wird dabei zwischen Symbolkommunikation bzw. Repräsentationstechniken und Affektkommunikation bzw. Präsentationstechniken unterschieden (Scheuermann 2009: 54–65).

Der Held **ICO** aus dem gleichnamigen Spiel (2001) muss ein Mädchen beschützen und sicher durch die Levels führen. Dazu nimmt er sie an die Hand.

Repräsentationstechniken

Techniken der Repräsentation basieren auf der Bedeutung der Dinge oder auch symbolischer Kommunikation, die es ermöglicht, mit Hilfe von Zeichen unter Bezug auf beliebige, zeitlich und räumlich nicht anwesende Gegenstände zu kommunizieren. Dies setzt notwendigerweise eine Decodierung und Interpretation durch den Rezipienten voraus und führt erst dann zu einer emotionalen Reaktion.

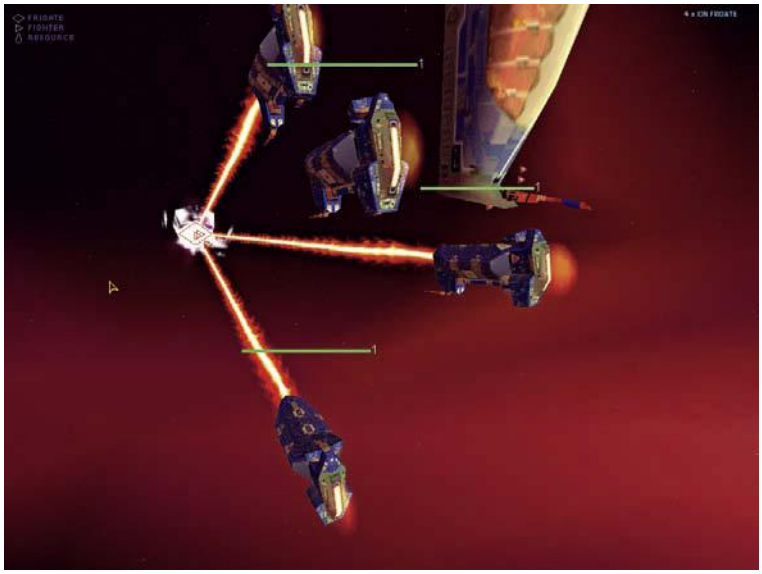
So hat zum Beispiel David Freeman unter dem Namen »Emotioneering« einen Ansatz entwickelt, der sich als Anwendung dramaturgischer Techniken auf elektronische Spiele darstellt, um glaubhafte Charaktere und Dialoge zu konstruieren, oder auch Spielwelt und Spielgeschichte mit erzählerischer Tiefe anzureichern. Bezeichnenderweise ist eine dieser Techniken dem Einbinden von Symbolen zur Steigerung der emotionalen Tiefe gewidmet (Freeman 2004).

Tatsächlich gehen diese Techniken jedoch kaum auf die medienspezifischen Besonderheiten elektronischer Spiele ein. Es gilt zwar, keine Angst vor Dramaturgen und ihren bewährten Techniken zu haben, da diese schon für Film und Fernsehen gute Ergebnisse produziert haben, doch es wäre fatal, »Interactive Narration« bzw. »Interactive Storytelling« als alleinige Lösung der Herausforderung einer reflektierten Anleitung zur Emotionalität in der Spielentwicklung zu sehen.

Walt Disneys abendfüllender Zeichentrickfilm **Bambi** (1942) hat viele Zuschauer zu Tränen gerührt.

En Taro Tassadar

Dass der Spieler lieb gewonnene Charaktere verliert, ist eine beliebte narrative Technik, um Emotionen zu wecken; so z.B. in den japanischen Rollenspiel-Serien Phantasy Star, Suikoden oder auch Final Fantasy. Hatte sich der Held Tassadar in Starcraft (1998) noch pathetisch geopfert, war der Spieler in der Erweiterung Broodwar (1998) gezwungen, gleich zwei Helden selbst zu vernichten. In jedem Fall muss die Umwelt auf den Tod der Charaktere reagieren, um den Spieler emotional anzuleiten, denn die Reaktion anderer wirkt oft stärker als die Sache allein.



In **Homeworld** (1999) suchen die letzten Überlebenden eines zerstörten Planeten nach einer neuen Heimat. Der Sound zu Beginn des Spiels ist affektiv perfekt inszeniert.

Statt sich wie David Cage oder David Freeman allein an den narrativen Mitteln des Films zu orientieren, ist es sinnvoller, sich der Affekttechniken zu bedienen, wie sie die Filmproduktion hervorgebracht hat.

Präsentationstechniken

Wer allein erzählerische Mittel einsetzen will, um Gefühlsregungen zu erzeugen, verkennt die Möglichkeiten elektronischer Spiele als audiovisuelle Medien. Und während Repräsentation immer auf Präsentation angewiesen ist, kann Präsentation auch ohne Bedeutungsebene wirken. Unter Präsentationstechniken sind solche Mittel zusammengefasst, die durch ihre reine Präsenz wirken und eben nicht über ihren Bedeutungsgehalt. Schon der antike römische Dichter Horaz wusste: »Mit den Lachenden lacht, mit den Weinenden weint das Antlitz des Menschen. Willst du, dass ich weine, so trauere erst einmal selbst; dann wird dein Unglück mich treffen [...].« (De arte poetica, 101)

Inzwischen weiß man, dass es wohl die sogenannten Spiegelneurone sind, die uns empathisch die Emotionen unseres Gegenübers allein durch deren Betrachtung mitfühlen lassen; die allein durch Beobachtung zugleich Vorstellungen anregen, Gedanken und Gefühle hervorrufen als auch körperliche Zustände verändern (Rizzolatti 2008). Dies hat nicht nur Auswirkungen auf die visuelle Darstellung, die in den letzten Jahren enorme Fortschritte gemacht hat, sondern betont auch die Bedeutung von Animation, Motion Capturing und Voice Acting sowie die Notwendigkeit, diese Aspekte frühzeitig im Produktionsprozess zu berücksichtigen. Es wird meist noch immer mehr Mühe darauf verwendet, die visuelle Realitätstreue von Schwerthieben und Blutgrätschen sicherzustellen, als darauf, eine überzeugende Wirkung der Gestik und Mimik von Gefühlsregungen zu entwickeln; mehr Zeit darauf, die Kollisionsabfrage von Nah- oder Fernkämpfen zu verbessern, als eine Umarmung oder auch nur das An-die-Hand-Nehmen einer Spielfigur glaubhaft darzustellen – vielleicht macht allein schon die Konzentration auf dieses Detail das Spiel ICO (2001) zu einem so großartigen Erlebnis.

All dies zusammen, ein Zittern in der Stimme oder ein markerschütternder Schrei, eine zaghafte Berührung oder eine feste Umarmung, ein sanftes Lächeln oder ein vor Trauer verzerrtes Gesicht, wird bereits mehr Affekte auslösen als das bloße Skript einer Geschichte allein.

Präsentationstechniken

- Wecken von Empathie durch ausdrucksstarke Animation, Motion Capturing und Voice Acting
- Anschauliches und detailliertes Vor-Augen-Stellen von Schlüsselreizen
- Anregen der Imagination durch undeutliche oder bloß kurz präsentierte audiovisuelle Reize
- Gezielter Einsatz von Sound/Musik zum Erzeugen von Affekten und Emotionen

Reize und Andeutungen

Audiovisuelle Präsentationstechniken initiieren Schlüsselreize wie die artspezifischen auslösenden Mechanismen (AAAM), von denen das Kindchenschema nur das Bekannteste ist. Der Körper reagiert auf viele dieser Reize, noch ehe die Kognition einzugreifen oder sich auf Abwehr einzustellen bereit ist. Auf die Züge eines kindlichen Gesichts, Formen von Schlangen und Spinnen oder auch die Farbe Rot reagiert der Körper des Betrachters unmittelbar und unwillkürlich (Eibl-Eibesfeldt 2004: 62–104). Ein in aller Deutlichkeit vor Augen gestellter Schlüsselreiz wird stets starke affektive Wirkungen erzielen. Einen ersten Katalog konventioneller filmisch-präsentativer Affekttechniken (etwa Lichtblitze oder laute Geräusche) hat Arne Scheuermann zusammengestellt (Ders. 2009). Daneben gibt es noch eine grundlegende Technik, die höchster Detailpräzision und Illusionismus gegenübersteht. Statt durch präsentative Deutlichkeit und Anschaulichkeit zu wirken, kann durch unklare oder dunkle Wahrnehmungen ein Reiz angedeutet werden, sodass er in der Imagination und Fantasie angeregt oder auch in Erinnerungen wachgerufen wird. In der Vorstellung können Reize dabei eine individuell intensivere Wirkung entwickeln als in der deutlichen Darstellung (Heinen 2004).

Musik als Affektlaut

Ebenso wie audiovisuelle Einzelreize lassen sich auch Aspekte der Musik mit ihren direkten und unbedingten Affektwirkungen zu den Präsentationstechniken zählen (De la Motte-Haber 1994). Dass mir insbesondere der Beginn des Spiels Homeworld (1999) als traurig in Erinnerung geblieben ist, ist vor allem dem Musikstück »Adagio for Strings« bzw. »Agnus Dei« von Samuel Barber zu verdanken. Ein Spiel mit einer Komposition zu unterlegen, die 2004 von

Schocktherapie

Das erste Spiel, das gezielt und in großer Menge Präsentationstechniken einsetzte, um den Spieler ohne Unterlass zu erschrecken, ist Doom (1993). Es affiziert den Spieler zugleich mit fließenden Bewegungen im Raum sowie mit detaillierten Splattereffekten und regt mit düsteren Sounds und Dunkelheit die Imagination an – so geschieht der Horror nicht nur auf dem Bildschirm, sondern auch im Kopf des Spielers. Diese Techniken, die für andere Affekte erst ausformuliert werden müssen, finden sich heute in beinahe allen Horrorspielen.

den Hörern der BBC zum »traurigsten klassischen Stück« gewählt wurde, mag wie ein recht billiger Trick wirken – doch er funktionierte ebenso u.a. bei Der Elefantenmensch (1980), Platoon (1986), Lorenzos Öl (1992) und Sophie Scholl – Die letzten Tage (2005). Die ungeheure Bedeutung, die der Musik also auch in Spielen zukommt, um Atmosphären zu schaffen und Affekte auszulösen, kann kaum genug betont werden. Auch hier kann man von den Techniken der Filmproduktion lernen.

Gameplay

Die Techniken der Repräsentation und Präsentation, welche oben beschrieben wurden, haben vor allem im Film ihr Zuhause und sind dort in den letzten Jahren immer weiter perfektioniert worden. Sich jedoch allein auf diese Mittel zu beschränken, missachtet die medienspezifischen Eigenschaften elektronischer Spiele und damit die Möglichkeit, sowohl Techniken aller Medien angemessen zu adaptieren als auch zugleich eigene Mittel zu entwickeln, die allein in diesem einen bestimmten Medium zu realisieren sind.

Wer wie David Cage die angestrebte Spielerfahrung mit einer Achterbahn vergleicht, läuft Gefahr, einer rein filmischen Präsentations- und Erzählweise zu verfallen und dabei unter dem Deckmantel der undefinierten »Interaktivität« eine Charakterisierung der Stärken des eigenen Mediums zu verfehlen. So kann es passieren, dass trotz aller Bemühungen nur ein weiteres Dragon's Lair (1983) entsteht – womöglich das beste Dragon's Lair aller Zeiten, aber eben doch nur ein Dragon's Lair. Es ist die eigentliche Herausforderung der Spielentwicklung der kommenden Jahre, in einer reflektierten Auseinandersetzung mit den Erkenntnissen und Errungenschaften anderer Medien einen eigenen, medienspezifischen Weg zu finden, der vor allem der totalen Einbindung des Spielers in das Geschehen, der vollständigen Immersionsillusion Rechnung trägt, wie sie eben einzigartig für elektronische Spiele ist.

Design vs. Emotion

Wie mühsam und doch gleichzeitig Erfolg versprechend diese Herangehensweise ist, zeigt der Entwicklungsprozess des Spiels Flower (2009), den Kellee Santiago im Rahmen der GDC Europe 2009 vorgestellt hat. Dabei wurde der gesamte Produktionsprozess sowie jede Designentscheidung an dem Ziel der Herstellung einer bestimmten Gefühlsregung ausgerichtet; was das angestrebte Spielerlebnis störte, wurde wieder eliminiert – ein Prozess, der viele Iterationen, Prototypen und Playtesting-Phasen erfordert sowie die Bereitschaft, gängige Herangehensweisen zu verwerfen bzw. immer neu zu ergänzen.

Im Gegensatz zu diesem Vorgehen kann klassisches Game Design eher störende Effekte entwickeln. Man hätte das Spielsystem von Flower auch komplexer gestalten und eine steigende Schwierigkeitskurve realisieren können. Doch dann wäre nicht dasselbe Ergebnis entstanden. Ein Spieler, der – im Sinne des Flow-Erlebens (Csikszentmihalyi 2000) – gerade vollständig darin aufgeht, ein durch das Game Design klar definiertes Ziel zu erreichen, die Werte seiner Ausrüstungsgegenstände gegeneinander abzuwägen oder eine strategische Kaufentscheidung zu treffen, wird sich seinen Gefühlsregungen leicht verschließen. Erst wenn sich Repräsentation, Präsentation und Gameplay dem intendierten Erleben unterordnen oder es zumindest als Designziel annehmen, werden Emotionen und Affekte in elektronischen Spielen systematisch zu erzeugen sein.

Entwicklung emotionalen Gameplays

- Konzentration auf die emotionale Wirkung bereits im Produktionsprozess und als Ziel des Designs
- Ein flexibler Produktionsprozess mit vielen Durchläufen, Prototypen und Testphasen
- Ggf. Verzicht auf die klassischen Ziele des Game Designs, um Affekte zu erreichen

Die Arbeit an sich selbst

Die problematische »Coolness«, die sowohl Spieler als auch Spielfiguren an den Tag legen, zeigt sich allerdings häufig bereits bei den Spieleentwicklern. Es bedarf in der Weiterentwicklung des elektronischen Spiels, schon allein der Effektsteigerung wegen, dringend der Selbstsensibilisierung für Gefühlsregungen, die über die in der Regel angestrebte »Coolness« hinausgeht. Um dieses Ziel zu erreichen, kann zum Beispiel ein System Anwendung finden, das Konstantin Stanislawski bereits Anfang des 20. Jahrhunderts unter dem Titel »Die Arbeit des Schauspielers an sich selbst« entwickelte und das heute unter dem Stichwort »Method Acting« bekannt ist (Stanislawski 2001). Vereinfacht zusammengefasst sollen Schauspieler mit dieser Methode zurückliegende Erfahrungen ih-

Der Klassiker **Dragon's Lair** (1983) von Readysoft gilt als Paradebeispiel eines interaktiven Films, der dem Spieler kaum Raum für Entscheidungen lässt. Stattdessen spult Dragon's Lair filmhafte Zeichentrick-Szenen ab.

Ethische Konsequenzen?

Wenn Spiele beginnen, etwas zu bedeuten und aufhören, »bloß Spiele« zu sein, stellen sich auch ethische Fragen nach der Inhaltlichkeit und den Möglichkeiten der Persönlichkeitsveränderung durch Spiele. Da Emotionen und Affekte unmittelbar Bedeutungen und auch Bewertungen erzeugen, Handlungen und Haltungen verändern, ist eine Spielentwicklung unter völliger Loslösung von ethischen Fragen kaum noch denkbar. Ist erst mal die Gesamtheit der Emotionen durch elektronische Spiele erreichbar und zugleich die Mechanik der Interaktion nicht mehr das wichtigste Ziel des Spieldesigns, so kann auch die für die ethische Entwicklung des Spielers bedeutende Emotionalisierung durch medienspezifische Mittel in greifbare Nähe rücken. Statt also die (zweifellos für viele Spieler faszinierende) Entemotionalisierung der Spielinhalte oder des Spielers anzustreben, stellt sich vielmehr die Frage, was man dem Spieler auf diesem direkten Wege vermitteln will.

Literatur

- A. Buchanan (2009): Facial Expressions for Empathic Communication of Emotion in Animated Characters. In: Animation Studies. (http://journal.animationstudies.org/2009/07/19/andrew-buchanan-facial-expressions-for-empathic-communication-of-emotion-in-animated-characters)
- M. Csikszentmihalyi (2000): Das Flow-Erlebnis. 8. Auflage. Stuttgart.
- A. R. Damasio (1999): Descartes' Irrtum. 4. Auflage. München.
- I. Eibl-Eibesfeldt (2004): Die Biologie des menschlichen Verhaltens. 5. Auflage. Vierkirchen-Pasenbach.
- D. Freeman (2004): Creating Emotion in Games. Boston u.a.
- U. Heinen (2004): Emotionales Bild-Erleben in der Frühen Neuzeit. In: Anthropologie der Literatur. Hrsg. von M. Engel et al. Paderborn, S. 356-383.
- U. Heinen (2007): Komponieren im Affekt. In: Wie sich Gefühle Ausdruck verschaffen. Hrsg. von Klaus Herding et al. Taunusstein, S. 161-188.
- U. Heinen (2008): Bildrhetorik der Frühen Neuzeit - Gestaltungstheorie der Antike. In: Design als Rhetorik. Hrsg. von Gesche Joost et al. Basel, S. 143-189.
- W. Kroeber-Riel et al. (2009): Konsumentenverhalten. 9. Auflage. München.
- H. de la Motte-Haber (1994): Die Musik als Affektlaut. In: Welt auf tönernen Füßen. Hrsg. von der Kunst- und Ausstellungshalle der BRD. Göttingen, S. 222-233, 458f.
- H. Mühlmann (2005): Maximal Stress Cooperation - MSC. Wien u.a.
- G. Rizzolatti et al. (2008): Empathie und Spiegelneurone. Frankfurt a.M.
- A. Scheuermann (2009): Zur Theorie des Filmemachens. München.
- K. S. Stanislawski (2001): Die Arbeit des Schauspielers an sich selbst, 2 Bände. Berlin.

res Lebens nochmals durchleben, die damals Affekte hervorriefen, die zu der vorliegenden Situation passen, und so spontane und emotionale Darstellung erzeugen. Die auf diese Weise generierten Affekte ermöglichen es dem Schauspieler, auf natürliche und unbewusste Veränderungen in Gesichtsausdruck, aber auch der ganzen Physiologie zurückzugreifen, wie Weinen, Stimmwechsel und Körpersprache. Hatten sich schon die Zeichner bei Walt Disney an der darstellerischen Leistung wirklicher Schauspieler orientiert, um ihre Charaktere zu entwickeln und mit Gefühlsregungen auszustatten, so wurde bei Pixar inzwischen eine Form des Method Acting eingeführt, um den Ausdruck der Figuren durch Eigenerfahrung zu verstärken (Buchanan 2009).

Emotionalität als Austausch

Schon die Maler der frühen Neuzeit wussten ihre Fantasie und Imagination gezielt durch transmediale Einflüsse anzuregen. Sie nutzten Texte, Dichtung, Lautmalerei und Musik, um im kreativen Prozess Gefühle und Vorstellungen auszulösen und als Impulse der Gestaltung einzusetzen. So konnten sie durch assoziatives Sammeln von Eindrücken (wie Exzerpte, Figuren oder Motive) ein anhand affektiver Hauptthemen geordnetes Musterbuch anlegen und selbstständig und psychologisch (glaubhaft) handelnde Figuren in der Bildhandlung imaginieren (Heinen 2007). Filme, Romane und Musik können heute ebenso gezielt eingesetzt werden, um sich für den medienspezifischen Gestaltungsprozess unter die Leitung von dem Design Document entsprechenden Emotionen und Affekten zu setzen. Ein bewusstes Erleben, Reflektieren und Sammeln derartiger Techniken ist die zwingende Voraussetzung, um diese auch kreativ und zielgerichtet einsetzen zu können.

Wer z.B. im Action-Rollenspiel **Diablo 2** (2000) nicht gerade die Aufregung angesichts eines Endgegners verspürt, geht komplett im Flow des Spiels auf.

Techniken der Selbstaffizierung

- Einsatz anderer Medien und persönlicher Erinnerungen, um sich selbst Emotionen oder Affekten auszusetzen
- Bewusstes Erleben, Reflektieren und Sammeln von bekannten und bewährten Techniken aus Film, Text oder Musik
- Bezug zu den Inhalten des Spiels entwickeln und diese im Kontext von Gefühlsregungen umsetzen

Hier liegt wohl die wahre Weisheit des römischen Dichters Horaz versteckt: Man muss immer erst mal selbst bewegt sein, bevor man sein Gegenüber bewegen kann. Es gehört also zu den dringlichsten Aufgaben der Spieleentwickler, die an der Weiterentwicklung ihres Mediums teilhaben wollen, sich erst einmal selbst neuen Emotionen und Affekten zu öffnen. Dies mag auch mit »Mach Spiele, die etwas bedeuten!« gemeint sein. Denn erst wenn man einen persönlichen emotionalen Bezug zum Gegenstand hergestellt hat und bereit ist, diesen zu teilen, können auch andere bewegt werden. Und – dies ist die zentrale Erkenntnis des anfangs zitierten Damasio – ohne Emotion gibt es überhaupt keine Bedeutung.

Björn Blankenheim

Das Spiel **Flower** (2009) für die PlayStation 3 versetzt den Spieler in der Rolle des Windes auf weite grüne Felder und erzeugt durch fließende Bewegungen und bunte Blütenblätter sanfte positive Emotionen.

Der Backstage-Pass der Spielebranche!

Jetzt **6x** im Jahr!

Making Games Magazin
Das Magazin für Spieleentwicklung und BusinessDevelopment

Profi-Wissen
Von Insidern für Insider

Profi-Chancen
Immer mit Firmenregister und Praktikumsbörse

Im Abo **30%** sparen!

Making Games Magazin erscheint im Verlag IDG Entertainment Media GmbH, Lyonel-Feininger-Str. 26, 80807 München, Handelsregister München, HRB 116413, Geschäftsführer: York von Heimburg. Die Kundenbetreuung erfolgt druch den Making Games Magazin Kundenservice, ZENIT Pressevertrieb GmbH, Postfach 810580, 70522 Stuttgart, Geschäftsführer: Peter Staudenmaier, Tel.: 0711/7252278, eMail: shop@makinggames.de

Bestellen Sie jetzt Ihren Backstage-Pass! Ihre Vorteile: 1) 30% gespart! 2) Bequeme Zustellung, pünktlich, portofrei!

6 HEFTE ☐ Ja, bitte schicken Sie mir das Making Games Magazin zum Jahresvorzugspreis von nur 28,98 €. Ich spare damit 30%. Die Lieferung erfolgt 6 x im Jahr und ist für mich vesandkostenfrei. Wenn ich 6 Wochen vor Ablauf des ersten Bezugsjahres nicht kündige, beziehe ich das Magazin ein weiteres Jahr.

1 HEFT ☐ Ja, bitte schicken Sie mir das nächste Making Games Magazin zum Einzelpreis von nur 6,90 € versandkostenfrei zu.

Firma ..... Name, Vorname .....

Straße/Postfach ..... PLZ ..... Ort .....

Telefonnummer ..... E-Mail ..... Fax .....

☐ Ich habe beruflich mit Spielen zu tun (Firma/Position) .....
☐ Ich bin interessierter Spieler
☐ Ich wünsche folgende Zahlungsweise ☐ bargeldlos und bequem per Bankeinzug ☐ per Rechnung

Bank ..... BLZ ..... Kto.-Nr. .....

9GM110A04J

Datum/Unterschrift .....

Infos & Bestellung:
www.gamestar.de/shopcode

Shop-Code 110
Shopcode merken für die Internet-Bestellung

oder
Tel.: 0711-7252 278
Mo-Fr 9-17 Uhr

Die Wiimote macht's möglich
Geheimnisse der Bewegungssteuerung

Egal ob Wii oder Natal: Bewegungserkennung ist bei Konsolenspielen längst ein wichtiges Thema. Wie man diese Technologie auch am PC nutzen kann, erklärt der Medieninformatiker Philipp Wollmann.

Philipp Wollmann

ist Student der Medieninformatik an der Universität Ulm.

Seit 2008 befasst sich Philipp mit Bewegungserkennung. Seine Diplomarbeit schrieb er über die Grundlagen der Nutzung von Bewegungen bei Anwendungen und Spielen. Seitdem entwickelt er für die Firma S.A.D. das YouMove-Bewegungssteuerungs-SDK für den PC.

WWW
makinggames.de

- Nintendo Wii: Chancen und Risiken der Bewegungssteuerung

Der große Erfolg von Nintendos Wii-Konsole ist zweifelsohne dem neuen Bedienkonzept und den darauf optimierten Spielen wie Wii Sports zu verdanken. Andere Hersteller wie Sony und Microsoft arbeiten mit Hochdruck an eigenen Lösungen für ihre Spielekonsolen Xbox 360 und PlayStation 3, um den Markt der bewegungsgesteuerten Spiele nicht allein Nintendo zu überlassen. Da aber auch der PC immer häufiger im Wohnzimmer an den HD-Fernseher angeschlossen wird und die klassischen Eingabegeräte Tastatur und Maus dort eher ungeeignet sind, gibt es mittlerweile PC-Bibliotheken für den Zugriff auf die Wiimote. Damit ist die Grundlage für ein neuartiges Bewegungs-Spielerlebnis am PC gegeben.

Wie man aus den Sensordaten letztendlich eine Bewegung abstrahiert, wieso das manchmal gar nicht so einfach ist und warum das selbst bei Nintendos Wii nicht immer richtig funktioniert, erklärt dieser Artikel.

Beschleunigung? Voll einfach! Oder?

Schauen wir uns einmal die Sensorwerte genauer an. Die Wiimote liegt gerade auf dem Tisch. Der Beschleunigungswert der x-Achse beträgt 0 g, die der y-Achse 0 g und die der z-Achse 1 g. Wir erinnern uns an den Schulunterricht in Physik: Das ist die Erdanziehungskraft, wir und alle Gegenstände beschleunigen zur Erde hin – damit haben wir schon das erste Problem. Die Erdanziehungskraft beeinflusst die Sensoren, und die Sensoren liefern im Ruhezustand nicht alle 0 g. Zumindest im Ruhezustand lässt sich durch ein wenig Trigonometrie die Ausrichtung der Wiimote im Raum ermitteln, denn die Erdanziehungskraft verteilt sich auf die drei Achsen der Beschleunigungssensoren. Aber was

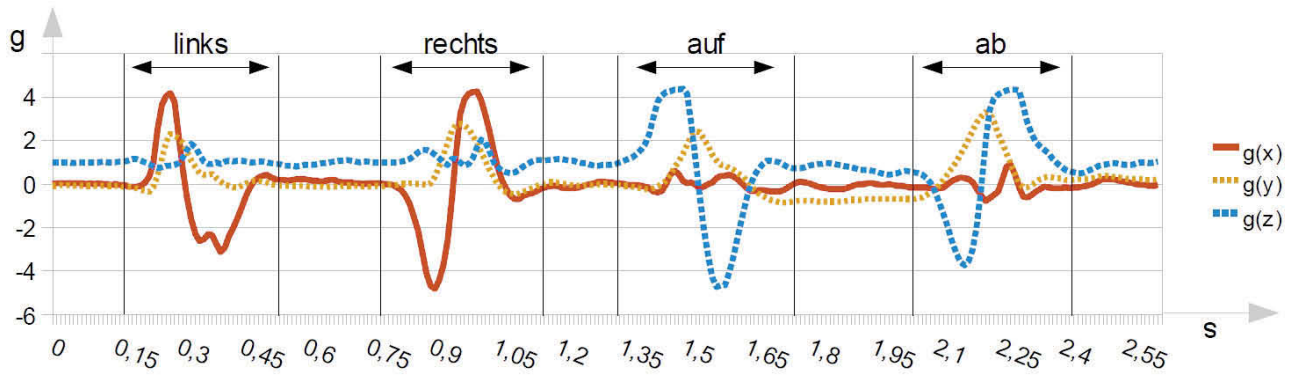


Abbildung 1: Graph der Wiimote 3D-Beschleunigungssensoren für eine Links-, Rechts-, Auf- und Abbewegung.

passiert, wenn wir die Wiimote bewegen? Genau – unsere Berechnung der Ausrichtung weicht von der tatsächlichen teils erheblich ab, da der Betrag der Beschleunigung der Wiimote nicht mehr 1 g ist. Das ist der Grund, wieso zum Beispiel der Baseball- oder Golfschläger in Wii Sports bei schnellen Bewegungen nicht das macht, was er sollte (und Wii MotionPlus kommen musste, um diesen Sachverhalt zu bessern).

Davon lassen wir uns jedoch nicht abschrecken und betrachten nun eine kurze Linksbewegung. Der Graph des X-Achsen-Sensors (rot) schlägt aus und ähnelt einer Sinusschwingung. Wieso das? Wir beschleunigen nach links und bremsen wieder ab, also beschleunigen wir während des Abbremsens nach rechts. Wir erinnern uns an den Physikunterricht, an die Trägheit der Masse. Der Graph einer Rechtsbewegung ist deshalb nur die Spiegelung der Linksbewegung. Analog dazu kann man Bewegungen nach oben und unten sowie vor und zurück erkennen.

Bewegungserkennung

Nun bauen wir uns einen Automaten, der die Links-rechts- und Auf-ab-Bewegungen anhand dieses Musters erkennt. Das funktioniert für einzelne Bewegungen nicht allzu schlecht, doch es ziehen schon wieder neue Wolken auf: Eine Links-rechts-Bewegung erzeugt keinen Graphen mit einer Sinusschwingung gefolgt von einer gespiegelten Sinusschwingung, sondern eine Sinusschwingung gefolgt von einer halben Sinusschwingung. Kratzen wir hier etwa an der Spitze des Eisbergs der Sonderfälle bei der Bewegungserkennung über Beschleunigungssensoren? Ja, genau das tun wir.

Man muss also entweder alle diese Sonderfälle berücksichtigen und intelligent erkennen, oder auf weitere Hilfsmittel wie die Sensorbar oder besser das aktuelle Wii MotionPlus zurückgreifen, das jedoch seinerseits neue Probleme mit sich bringt.

Mensch gegen Maschine

Schauen wir genauer hin. Wieso schlägt bei einer sauberen Linksbewegung nicht nur die x-Achse aus, sondern auch in hohem Maße die y-Achse? Weil Menschen keine Roboter sind und nicht nur links/rechts, vor/zurück, hoch/runter, sondern auch alle Nuancen dazwischen kennen. Speziell nutzen wir bei der Wiimote das Handgelenk und nicht den ganzen Arm, wodurch auf der y-Achse zwangsweise starke Fliehkräfte entstehen.

Weiter entsteht bei einem scheinbar sauberen Schlag nach unten meist immer ein starker Ausschlag auf der x-Achse. Das ist durch die Muskulatur zu erklären, denn bei einer Abwärtsbewegung helfen mehrere Muskeln, und die führen anders als bei Robotern zu Interferenzen, die man selbst jedoch nicht wahrnimmt. Diesem Problem muss man sich bei der Bewegungserkennung durch Beschleunigungssensoren stellen – mit all seinen komplexen Folgen. Neben der Maskierung muskulärer Interferenzen bringt Wii MotionPlus hier eine erhebliche Verbesserung.

Bewegungserkennung, die zweite

Als Spieleentwickler möchte man reale Bewegungen mit der Wiimote möglichst 1:1 auf Bewegungen im virtuellen Raum umsetzen. Nehmen wir als Beispiel ein Jump & Run. Hier hätten wir gern, dass eine reale Bewegung nach rechts oben den kleinen Mann im Monitor ebenfalls mit der Stärke des Schlages dementsprechend stark nach rechts oben springen lässt. Dazu müssen wir also aus den Beschleunigungswerten den zurückgelegten Weg (oder auch nur die Geschwindigkeit: einfaches Integral) berechnen. Das wäre das zweifache Integral über die Beschleunigung. Ab hier wird klar, welchen Umfang die Bewegungserkennung hat. Denn die Latenz wäre zu groß, wenn man das Integral über die komplette Bewegung bilden würde. Latenzfrei fühlt es sich nur an, wenn man beim ersten Minimum oder Maximum der Beschleunigung den Sprung des Männchens im Monitor ausführt. Folglich muss man drei Viertel der Strecke interpolieren. Die entstandenen Fehler halten sich dabei jedoch in Grenzen. Allerdings stößt man dann auf das nächste Problem. Wenn man die Wiimote nicht waagerecht zum Boden hält, weicht die Basis von der x- und z-Achse nämlich von 0 g beziehungsweise 1 g ab und man muss je nach Ausrichtung der Wiimote das Integral über eine dynamische Basis bilden. Vom Spieler kann man nicht erwarten, dass er die Wiimote stets waagerecht hält, daher

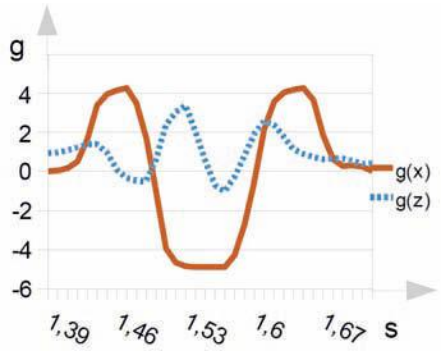


Abbildung 2: Graph bei Links-rechts-Bewegung.

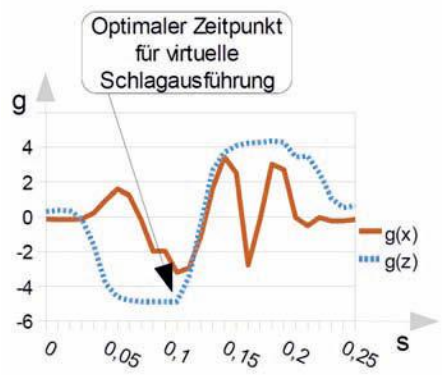


Abbildung 3: Graph bei starkem Abwärtsschlag.

sind diese Schritte zwingend erforderlich. Doch wo fängt dieses Integral an und wo hört es auf?

Spätestens beim Betrachten der Beschleunigungswerte beim wilden Umherfuchteln mit der Wiimote wird klar, dass Bewegungserkennung eine echte Herausforderung ist, denn die Graphen sind nicht zuletzt wegen der geringen Auflösung von ± 3 g im 10-ms-Intervall oft mehrdeutig. Eine Verbesserung verspricht eine ausgeprägte Analyse und Einbindung der zahlreichen Sonderfälle in den Algorithmus, denn der Referenzpunkt Erdanziehungskraft schwindet mit der Stärke der Bewegung zusehends.

Geeignete Hilfsmittel

Ein alternativer Referenzpunkt für schnelle Bewegungen wäre die Sensorbar oder eine Kamera, die die aktuelle Position der Wiimote im Raum erkennt. Das schränkt allerdings wiederum die Bewegungsfreiheit ein. Deutlich einfacher funktioniert das mit Wii MotionPlus, was die Bewegungsfreiheit nicht einschränkt und nahezu eine 1:1-Abbildung verspricht. Doch auch hier summieren sich kleine Fehler. Abweichungen müssen nachkalibriert oder intelligent berichtigt werden. Nintendo hat sich bei der Wii für die Lösung entschieden, die Wiimote mit dem Motion-Plus-Addon immer wieder nachzukalibrieren, indem man sie verkehrt herum auf den Tisch legt. Bei Spielen wie Wii Sports Resort spürt man deutlich, dass die Sensorbar ein Referenzpunkt ist, denn man muss die Wiimote anfangs nach vorn richten, damit die Wii weiß, dass der Ausgangspunkt vor der Brust ist. Ab dann summieren sich kleine Fehler, und die virtuelle kann von der realen Bewegung abweichen.

Wahrscheinlichkeitsmodelle, die mögliche Bewegungen des Menschen berücksichtigen, helfen dabei, eine möglichst realistische Abbildung zu erfassen. AiLive hat dies zum Beispiel bei der Entwicklung von Wii MotionPlus gemacht. Sehr viele dieser kleinen Tricks sind nötig, um eine vernünftige Bewegungserkennung auf den Weg zu bringen. Vor allem die kritischen Latenzzeiten erfordern eine ausgeklügelte Bewegungserkennung, die auf die Besonderheiten des Menschen eingeht.

Fazit

Eine Wii-Konsole findet man heute in sehr vielen Haushalten, wodurch eine Bewegungssteuerung am PC meist ohne weitere Hardware-Anschaffungen möglich ist. Wegen der hohen Komplexität der Bewegungserkennung durch die Tücken der menschlichen Motorik und der besonderen Eigenschaften von Beschleunigungssensoren bilden freie C++-Bibliotheken wie WiiYourself oder WiiUse nur die Grundlage, um auf die Sensoren der Wiimote zugreifen zu können. Weitere zeitaufwändige Schritte sind allerdings notwendig, um die Steuerung für Spiele nutzen zu können.

Wenn die Grundlagen für die Wohnzimmer-PC-Steuerung gegeben sind, besteht die Hoffnung, dass der PC nicht von den Konsolen verdrängt wird. Wie bei den Konsolen hängt auch beim PC der Erfolg stark von der Steuerung ab. Ein neuer Lifestyle bezüglich Computer und Unterhaltungselektronik erfordert auch am PC ein Umdenken.

Philipp Wollmann

Links

YouMove SDK (PC-Bewegungssteuerung)
www.you-move.de

Wiimote Basics
http://wiibrew.org/wiki/Wiimote

Wiimote APIs
http://wiiyourself.gl.tter.org
www.wiiuse.net
www.codeplex.com/WiimoteLib

3D-MÄUSE FÜR INGENIEURE,
DESIGNER UND ARCHITEKTEN

3Dconnexion | NAVIGATE YOUR 3D WORLD

Es ist, als würden Sie in den Bildschirm greifen und Ihre 3D-Modelle mit der Hand bewegen.

SpaceNavigator™

Ihr perfekter Einstieg in die Welt der 3D-Mäuse:

Stellen Sie sich vor, Sie könnten ein 3D-Modell auf Ihrem Monitor so einfach steuern, als würden Sie es in Ihrer Hand halten. Oder fliegen Sie wie mit einem Helikopter durch dreidimensionale Welten wie Google Earth. Mit einer 3D-Maus von 3Dconnexion erleben Sie eine völlig neue Dimension der 3D-Navigation.

Weitere Informationen zur 3D-Maus-Produktfamilie | www.3dconnexion.de

Noten- statt Wellensurfen
Musikspiel mal anders

Drei Studenten der Berliner Hochschule für Technik und Wirtschaft sagen Guitar Hero und Co den Kampf an. Mit einem Skateboard.

Stephan Krohn

ist studentische Hilfskraft im gamelab der HTW Berlin.

Stephan ist einer von drei Studenten der Berliner HTW, die an Music Racer arbeiten. Nach einer zweijährigen Ausbildung zum Audio Engineer an der SAE Köln studierte er Internationale Medieninformatik und machte seinen Master mit dem Schwerpunkt Game Technology.

WWW
makinggames.de

- Noch mehr Virtual Reality: das Studentenprojekt Space Trash

Ausgangspunkt des von uns Music Racer genannten Projekts waren die Kurse »Games and Simulation Technology« und »Realtime Interactive Systems«. Diese Lehrgänge werden im Rahmen des Masterprogramms »Internationale Medieninformatik« im Schwerpunkt »Game Technology and Interactive Systems« an der Hochschule für Technik und Wirtschaft Berlin (HTW) angeboten. Die Aufgabe war es, mit einer kleinen Gruppe innerhalb eines Semesters ein eigenes Computerspiel zu entwickeln. Dabei sollte möglichst keine komplette Game Engine verwendet werden, sondern einzelne Komponenten miteinander kombiniert werden. Das daraus entstandene Spielkonzept wurde anschließend in einem Folgeprojekt wieder aufgegriffen und um einige immersive Komponenten erweitert.

Die Idee

Die Gruppe war sich recht schnell darüber einig, diese Gelegenheit zu nutzen, um die Möglichkeiten von musikbasierten Spielen auszuprobieren. Zu den Vorreitern in diesem Segment zählen Guitar Hero von Activision Blizzard und Electronic Arts' Rock Band. Im Unterschied zu den beiden AAA-Titeln sollte unser Spiel allerdings zwei grundlegende Änderungen enthalten:

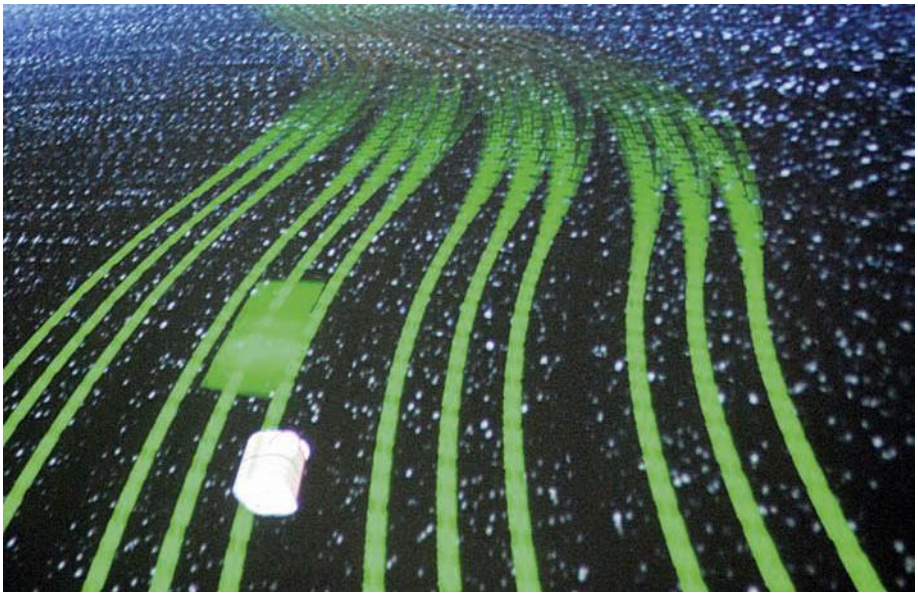
1. Guitar Hero und Rock Band versuchen das Musizieren möglichst realitätsnah wiederzugeben. In Music Racer hingegen sollte das Steuerungskonzept auf Rhythmus basieren.
2. Während bei den Vorbildern der Spielinhalt im Vorfeld erstellt werden muss, sollte dieser nun dynamisch aus der Musik selbst generiert werden.

Als Thema des Spiels einigten wir uns auf ein Rennspiel, bei dem die Strecke aus einem beliebigen Audiofile generiert werden sollte.

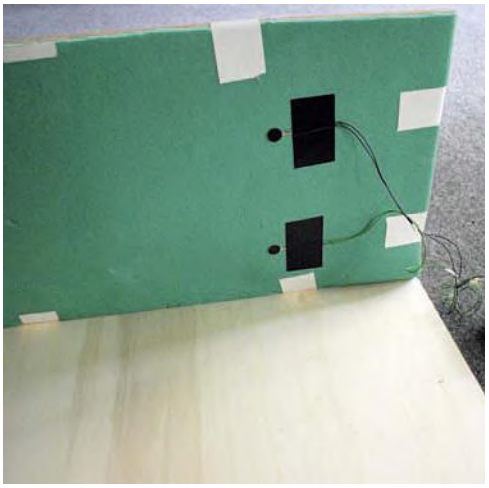
Die Planung

Da die Umsetzung des Spiels viel Forschung erforderte, war die Entwicklungszeit von knapp zwei Monaten für ein Team von drei Studenten, die gleichzeitig noch andere Kurse belegt hatten, sehr knapp. Aus diesem Grund musste die erste Version von Music Racer grafisch schlicht gehalten werden. Zu diesem Zweck wurde das Szenario in eine futuristische Umgebung (Weltraum) verlegt, was zudem den Vorteil mit sich bringt, dass man sich nicht sklavisch an die Realität halten muss und selbst bestimmen kann, wie etwas aussieht. Um eine möglichst unabhängige und reibungslose Zusammenarbeit zu gewährleisten, wurde das Projekt in drei Teile

Musikspiel und Balance-Übung in einem: **Music Racer** steuert man mit einem Skateboard.



Die **Strecken** werden nicht von den Entwicklern gebaut, sondern aus der vom Spieler gewählten Musik heraus generiert. Dabei ist es egal, ob Sie Klassik, Rockmusik oder Pop-Melodien verwenden.



Das Raumschiff wird durch Gewichtsverlagerung gelenkt. Das **Skateboard** ist dabei auf einem provisorischen Brett fixiert, das mit **Drucksensoren** ausgestattet ist, wie man sie von Waagen kennt.

gegliedert: die Audioanalyse, den Streckenbau und das sogenannte Framework (GUI, Optionen etc.). Diese Planung führte zu einer optimalen Ausnutzung der zur Verfügung stehenden Entwicklungszeit. Ein weiterer positiver Nebeneffekt ist der dadurch entstandene modulare Aufbau: Die Komponenten sind komplett unabhängig voneinander, weshalb beispielsweise die Audioanalyse problemlos auch in anderen Spielen eingesetzt werden könnte.

Das Framework

Der nächste Schritt bestand darin, geeignete Komponenten für die Game Engine auszuwählen. Die wichtigsten Anforderungen an die Komponenten waren:

1. einfache Integration
2. einfache Benutzung
3. Funktionsumfang

Wir wählten die Game Engine »Horde3D«. Entscheidend waren hier vor allem die Kompaktheit und die dadurch einfache Handhabung. Für die Audio-Komponente fiel die Wahl auf »Fmod-Ex«, eine weit verbreitete Lösung, die fast alle für unsere Zwecke benötigten Funktionen bietet. Zudem erlauben die Lizenzbestimmungen die freie Nutzung in nicht kommerziellen Projekten. Als größtes Problem erwies sich überraschenderweise die Benutzeroberfläche (GUI). Hier fielen die meisten Lösungen bereits wegen der extrem umständlichen Einbindung durch das Raster. Letzten Endes entschieden wir uns für »guichan«. Für Input Management und Window Management wurde »SDL« verwendet. Auf das Einbinden einer Physik-Komponente (etwa Ageia PhysX) wurde indes verzichtet, da es in Music Racer nicht nötig war, komplexe physikalische Vorgänge zu simulieren.

Die Umsetzung

Das Konzept, das hinter dem dynamischen Streckenbau steckt, lässt sich wohl am besten als »Carrera-Bahn-Prinzip« beschreiben. Die Idee dabei war es, die Strecke aus Einzelteilen zusammenzusetzen, die wiederum aus der Musik generiert werden. Dazu wird zunächst das Audiomaterial in eine Sekunde lange Abschnitte aufgeteilt. Anschließend werden zwischen diesen Abschnitten Unterschiede im Klangbild ermittelt, wobei es besonders auf das Stereo-Panorama, den Pegel und den Rhythmus des Liedes ankommt. Diese Daten werden dann an das Streckenbau-Modul geschickt, welches die Elemente mit entsprechender Steigung bzw. Kurvengrad generiert. Zusätzlich werden aus den Informationen der Beat-Analyse Hindernisse und Boni erzeugt, die auf der Strecke verteilt werden und nachher vom Spieler umfahren oder eingesammelt werden müssen.

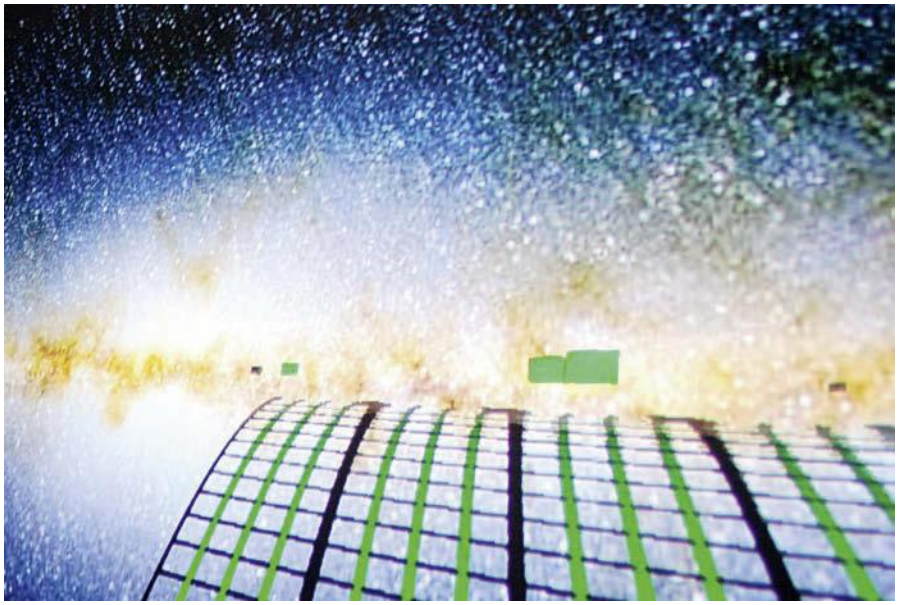
Das Gameplay

In Music Racer wählt der Spieler zunächst eine beliebige Audiodatei. Aus dieser wird eine Strecke berechnet, über die man schließlich das Fahrzeug steuert. Die Steuerung erfolgt wahlweise über die Maus oder einen speziell entwickelten Controller. Der Streckengenerator ist deterministisch; es gibt also für jeden Song einen individuellen Kurs. Insgesamt bietet Music Racer vier verschiedene Spielmodi, in denen entweder auf Zeit oder nach Punkten gespielt wird. Dabei nehmen die Hindernisse jeweils andere Funktionen ein (Punkteboni oder -abzüge, schneller/langsamer werden).

Die Audio-Analyse

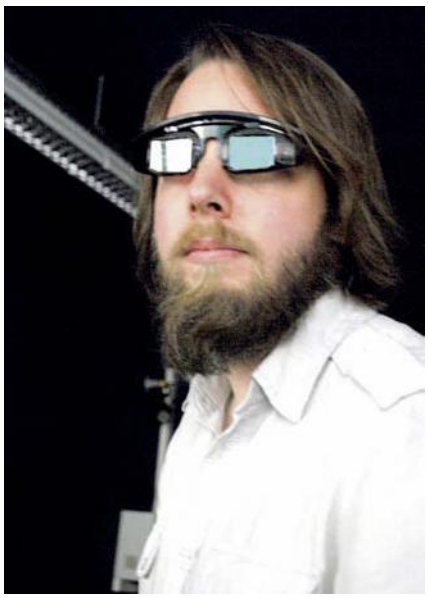
Das Herzstück des Spiels ist die Analyse der Audiodaten. Zusätzlich zur Prüfung der einzelnen Abschnitte führt das Programm noch eine globale Beat-Analyse durch. Hierzu wird das Signal mithilfe von FMOD zunächst durch eine Fast Fourier Transformation (FFT) in Frequenzspektren zerlegt und anschließend in Frequenzbändern zusammengefasst, die wiederum auf Maxima untersucht werden. Zeitabschnitte, in denen besonders große Maxima auftreten (die also über einem einstellbaren Threshold liegen), werden als Beats markiert. Somit lässt sich über einen einzigen Parameter bereits eine Anpassung des Schwierigkeitsgrades vornehmen, da

Das **Team** von Music Racer: Dennis Lübke, Stephan Krohn und Benjamin Hergert.



Rhythmische Elemente in den Songs erzeugen **Hindernisse** und Boni auf den Strecken.

ein niedrigerer Threshold zu mehr erkannten Beats und somit zu mehr Hindernissen führt. Ein weiterer wichtiger Punkt war die Verknüpfung von Song- und Streckenposition. Dies ist zum einen für die Bewegung auf der Strecke und zum anderen für die exakte Platzierung der Hindernisse wichtig. Da das Signal für die FFT ohnehin in Buffer aufgeteilt werden musste, wurden diese Buffer zudem als »Waypoints« für die Lokalisation genutzt. Solange die Buffer klein genug sind, gewährt unser Prinzip sowohl eine exakte Positionierung als auch einen flüssigen Bewegungsablauf. Zusätzlich zu einer perfekten Synchronisierung von Bild und Ton, die bei einem solchen Spiel elementar wichtig ist, ermöglichte dies zudem den interessanten Nebeneffekt, die Renngeschwindigkeit über die Abspielgeschwindigkeit der Musik zu verändern. Das wird beispielsweise im Modus »Race« ausgenutzt, bei dem Spiel und Song durch Kollisionen mit Hindernissen gebremst werden.



Mit sogenannten **Shutter-Brillen** lässt sich Music Racer auch in echtem 3D spielen.

Die Immersion

Die erste Version von Music Racer benutzte die Maus als primäre Steuerung. In einem Projekt mit dem gameslab der HTW Berlin, das sich mit dem Einsatz von Sensorik in Spielen beschäftigte, entstand jedoch die Idee, das Programm immersiver zu machen. Zu diesem Zweck wurde ein eigener Controller entwickelt, der dem Spieler das Gefühl vermitteln sollte, tatsächlich über die Strecke zu surfen. Dieser Controller besteht aus einem alten Skateboard, unter dem Sensoren befestigt sind, die die Gewichtsverteilung auf dem Board messen, vergleichbar mit dem Wii Board. Der Unterschied ist allerdings, dass das Skateboard zwar an einer Stelle fixiert ist, eine Gewichtsverlagerung aber nach wie vor eine Neigung des Boards zur Folge hat. Dies vermittelt ein realistischeres Spielgefühl, da der Spieler direkt Rückmeldung bekommt und zusätzlich zum Spielgeschehen auch auf seine eigene Balance achten muss, um nicht vom Board zu fallen. Die verwendeten Sensoren sind einfache Drucksensoren, wie sie etwa in handelsüblichen Waagen zum Einsatz kommen. Um Music Racer noch intensiver zu gestalten, wurde die Spielgrafik auf die sogenannte Stereo-Technologie umgestellt, sodass eine 3D-Projektion im Polarisationsverfahren möglich ist. Durch die dreidimensionale Projektion wird das Gefühl des Spielers, tatsächlich durch die Steilkurven zu surfen, noch einmal verstärkt; Strecke und Hindernisse wirken nun sehr plastisch. Um das zu erreichen, mussten wir eine zweite Kamera ins Spiel einbauen, deren Perspektive im Vergleich zur Hauptkamera leicht verschoben ist, was den natürlichen Augenabstand des Menschen simuliert. Das Problem der Polarisationstechnik ist allerdings, dass hierfür bestimmte Projektoren bzw. Aufbauten und Grafikkarten notwendig sind und der Benutzer eine spezielle Shutter-Brille tragen muss. Ein entsprechender Rechner mit Projektor wurde vom gameslab zur Verfügung gestellt.

Das Fazit

Mit einem kleinen Team aus drei Studenten und sehr begrenzter Entwicklungszeit (insgesamt drei Monate) ist ein Spiel entstanden, das intuitiv spielbar ist und eine breite Zielgruppe anspricht. Music Racer wurde bereits einige Male im gameslab präsentiert und konnte von den Besuchern getestet werden. Dabei fiel auf, dass das Spiel durch alle Altersgruppen hinweg begeistert gespielt wurde. Auch ein Unterschied zwischen den Geschlechtern war nicht auszumachen, was vor allem an der intuitiven Kontrollmethode liegen dürfte. Ein weiterer Grund ist, dass das Spiel mit jeder Art von Musik funktioniert. So war festzustellen, dass sich zu den Tönen von den Beatsteaks eher junges Publikum um den Stand scharte, während bei entsprechender Musikwahl auch die ältere Generation zur Testfahrt antrat.

Der Ausblick

Music Racer wird kontinuierlich weiterentwickelt. Im Moment wird neben Verbesserungen im grafischen Bereich vor allem an der Portierung auf andere (mobile) Plattformen gearbeitet.

Durch den eingangs erwähnten modularen Aufbau der Software wäre es außerdem denkbar, das Prinzip der audiobasierten Generierung von Content auf andere Genres und Spielmechaniken anzuwenden. Das ist zurzeit eines der Forschungsthemen des gameslabs.

Stephan Krohn

making games TALENTS

DEUTSCHLANDS TALENTBÖRSE FÜR SPIELE-ENTWICKLER

DEIN WEG IN DIE SPIELEBRANCHE!

Acht Entwicklerstudios suchen die Macher von morgen. Du willst gefunden werden? Dann komm am 24. Oktober ins IDG-Medienhaus* in München!

Anmeldeinformationen unter www.makinggames.de/talents

* Lyonel-Feininger-Straße 26 · 80807 München

ARTIFICIAL · RADON LABS · CipSoft online entertainment · sproing · SevenOne Intermedia · BIGPOINT · CRYTEK · TRAVIANGAMES

WWW
makinggames.de
- Online-Praktikumsbörse mit Suchfunktion

Berufschance Spiele-Industrie
Praktikumsbörse

Sie studieren Informatik, möchten aber lieber Spiele statt Büro-Software programmieren? Sie kennen sich mit Maya aus und wollen Ihr künstlerisches Talent zum Beruf machen? Hier ist Ihre Chance: Spiele-Entwickler und Publisher aus ganz Deutschland suchen Praktikanten.

Praktikanten gesucht?

Sie suchen motivierte und talentierte Praktikanten? **info@makinggames.de, Betreff: Praktikum**
Der Abdruck ist kostenfrei.

PLZ-BEREICH 10000

Exozet Games
Art des Praktikums: Nach Absprache
Bereich: Maya 3D Max Modelling, Animation
Mindestdauer: 6 Monate
Ansprechpartner: Bartol Ruzic, bewerbung@exozet.com
Adresse: Rotherstraße 20, 10245 Berlin

Fraunhofer IGD Rostock
Art des Praktikums: Nach Absprache
Bereich: Prototyping, Modellierung, Programmierung, User Interface Design, Charakterdesign/Illustration, Sound- Akustikdesign
Mindestdauer: 3 Monate
Ansprechpartner: Holger Diener, holger.diener@igd-rfraunhofer.de, Katharina Schmidt, katharina.schmidt@igd-rfraunhofer.de
Adresse: J.-Jungius-Str. 11, 18059 Rostock

Future Interactive
Art des Praktikums: Studienbegleitend
Bereich: Java-Programmierung, C++-Programmierung
Mindestdauer: Nach Absprache
Ansprechpartner: Felix Kollmann, fkollmann@futureint.de, 030 / 810 593 36
Adresse: Gartenstr. 1, 14169 Berlin

GameDuell
Art des Praktikums: Studienbegleitend, nach Absprache
Bereich: Programmierung, Marketing, Controlling/Finanzen
Mindestdauer: 6 Monate
Ansprechpartner: Tina Haile, jobs@gameduell.de, 030 / 288 768 211
Adresse: Ackerstr. 14-15, 10115 Berlin

Radon Labs
Art des Praktikums: Studienvorbereitend, studienbegleitend, Praxis-Semester, Diplom-Semester
Bereich: Programmierung, Programmierung (3D, AI, Physik), 3D-Grafik, Animation, Maya Modelling, Leveldesign
Mindestdauer: 3 Monate, besser 5-6 Monate
Ansprechpartner: Bernd Beyreuther, bernd@radonlabs.de
Adresse: Bouchéstraße 12, 12435 Berlin

Silver Style Entertainment
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Scripting (Python)
Mindestdauer: 6 Monate
Ansprechpartner: Carsten Strehse, carsten.strehse@silver-style.com
Adresse: Am Borsigturm 12, 13507 Berlin

Yager Development
Art des Praktikums: Vollzeit, Praktikumssemester, studienbegleitend, praktische Diplomarbeit
Bereich: Programmierung (AI, 3D, Physik), Grafik (2D/3D), Scripting, Animation, Leveldesign
Mindestdauer: 3 Monate
Ansprechpartner: Eike Stolley, eike.stolley@yager.de
Adresse: Pfuelstr. 5, 10997 Berlin

PLZ-BEREICH 20000

Bigpoint
Art des Praktikums: Vollzeit
Bereich: Produktanalyse/Produktmanagement, Gamedesign
Mindestdauer: 3 Monate
Ansprechpartner: Human Resources, personal@bigpoint.net, Betreff bitte »Praktikumsbörse, Produktanalyse« oder »Praktikumsbörse, Gamedesign«
Adresse: Christoph-Probst-Weg 3, 20251 Hamburg

Borgmeier Media Communication
Art des Praktikums: Praktikum für Studenten mit Schwerpunkt Marketing
Bereich: Games-PR, Corporate-PR, Marketing
Mindestdauer: 3 Monate
Ansprechpartner: Tanja Deilecke, deilecke@agentur-borgmeier.de, 040 / 413 09 60
Adresse: Rothenbaum chaussee 5, 20148 Hamburg

CNT Media
Art des Praktikums: Nach Absprache (Übernahme möglich)
Bereich: Kommunikation, Level-Design, Projekt-Planung
Mindestdauer: 3 Monate
Ansprechpartner: Cornel Hillmann, ch@cnt-media.net
Adresse: Süderstr. 159a, 20537 Hamburg

Daedalic Entertainment
Art des Praktikums: Vollzeit
Bereich: Programmierung, Grafik, Marketing
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Dominik Meyer, dm@daedalic.de, 040/ 432 612 72
Adresse: Papenreye 53, Workport Unit 2, 22453 Hamburg

dtp entertainment
Art des Praktikums: Nach Absprache
Bereich: PR, Marketing, Grafik, Test, Produktmanagement
Mindestdauer: 6 Monate
Ansprechpartner: Claas Wolter, c.wolter@dtp-ag.com, 040 / 669 910 43
Adresse: Goldbekplatz 3-5, 22303 Hamburg

House of Tales Bremen
Art des Praktikums: Vollzeit
Bereich: C++/Delphi-Programmierung, Scripting, Implementation, Integration, 3DSMAX, Rendering, Low-Poly-Modelling, In-Game-Dialog, Cinematics Scripting, Storytelling, Design-Dokumentation
Mindestdauer: 3 Monate
Ansprechpartner: Martin Ganteföhr, martin@house-of-tales.com
Adresse: Dobbenweg 3, 28203 Bremen

Indigo Pearl
Art des Praktikums: Nach Absprache
Bereich: PR-Agentur
Mindestdauer: 6 Monate
Ansprechpartner: Torsten Oppermann, to@indigopearl.de, 040 / 253 302 67
Adresse: Süderstr. 159a, 20537 Hamburg

Innogames
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Software-Entwicklung (PHP), Web- & Grafik-Design, Systemadministration (Linux), 3D-Modelling
Mindestdauer: 6 Wochen
Ansprechpartner: Michael Zillmer, info@innogames.de, 04141 / 776 177
Adresse: Theodor-Haubach-Weg 2, 21684 Stade

northworks
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Grafik (2D/3D), Programmierung PHP, Programmierung Flash ActionScript, Community-Management (mehrsprachig)
Mindestdauer: 3 Monate
Ansprechpartner: Sascha Kaddatz, jobs@northworks.de, 040 / 781 070 75
Adresse: Stahltwiete 23, 22761 Hamburg

SFK Spielekanal Fernsehen
Art des Praktikums: Vollzeit, nach Absprache
Bereich: u.a. Game-Entwicklung und Programmierung, Game Konzeption und Design, iTV, IPTV, DVB-H im Spiele-Kontext
Mindestdauer: 3 Monate
Ansprechpartner: Claudia Marloh, claudia.marloh@spika.tv, 040 / 348 094 974
Adresse: Alter Wandrahm 15, 20457 Hamburg

SnapDragon Games
Art des Praktikums: Nach Absprache
Bereich: Grafik, (Modelling, Leveldesign, Basis Maya), Programmierung (3D, AI, Physik, Tcl)
Mindestdauer: 3 Monate
Ansprechpartner: Christian von Duisburg, hr@snapdragongames.com
Adresse: Schulterblatt 58 , 20357 Hamburg

Spacetale
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: 2D/3D-Grafik
Mindestdauer: 3 Monate
Ansprechpartner: Michael Wolf, jobs@spacetale.de
Adresse: Boppstr. 26, 55118 Mainz

Spotsonfire
Art des Praktikums: Nach Absprache
Bereich: Marketing, Programmierung, 2D-Design, Illustration, Animation
Mindestdauer: 3 Monate
Ansprechpartner: jobs@spotsonfire.com, 040 / 209 346 330
Adresse: Hongkongstraße 7, 20457 Hamburg

Ticking Bomb Games
Art des Praktikums: Vollzeit
Bereich: Programmierung, Grafik
Mindestdauer: 3 bis 6 Monate
Ansprechpartner: Marco Schultz, jobs@ticking-bomb-games.de
Adresse: Adenauerallee 32, 20097 Hamburg

PLZ-BEREICH 30000

Reakktor Media
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik, Web-Programmierung, Leveldesign
Mindestdauer: 6 Monate
Ansprechpartner: Kirk Linke, linke@reakktor.com, 0511 / 540 980
Adresse: Arnswaldtstrasse 1030159 Hannover

Rocketbox Studios
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik
Mindestdauer: 5 Monate
Ansprechpartner: Markus Wojcik, jobs@rocketbox.de, 0511 / 935 72 80
Adresse: Vahrenwalder Str. 7, 30165 Hannover

Xybris Interactive
Art des Praktikums: Nach Absprache
Bereich: Programmierung (Java, PHP), Grafik, Webdesign (CBS)
Mindestdauer: 3 Monate
Ansprechpartner: Christian Sauer, info@xybris.com
Adresse: Apfelstr. 12a, 33613 Bielefeld

PLZ-BEREICH 40000

bhv Software
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Produktmanagement, Games, Non-Games, Edutainment, Herstellung, PR
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Dedeke, bewerbung@bhv.de, 02131 / 765 01,
Adresse: Novesiastr. 60, 41564 Kaarst

Ubisoft / Blue Byte
Art des Praktikums: Nach Absprache
Bereich: Programmierung, Game Design, Level Design, Grafik, Projektmanagement
Ansprechpartner: siehe unter http//job.ubisoft.com/MAIN/careerportal, 0211 / 338 000
Adresse: Adlerstr. 74, 40211 Düsseldorf

Zone 2 media
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Programmierung, 3D-Grafik
Mindestdauer: 6 Monate, nach Absprache
Ansprechpartner: Holger Offermanns, h.offermanns@zone2.de, 02150/70566-0
Adresse: Am Wasserturm 6, 40668 Meerbusch

PLZ-BEREICH 50000

Bright Future
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: C++-Programmierung
Dauer: 4 Monate, besser 6 Monate
Ansprechpartner: Gerald Köhler, jobs@brightfuture.de, 0221 / 789 821 71
Adresse: Im Zollhafen 15-17, 50678 Köln

EA
Art des Praktikums: Vollzeit
Bereich: PR, Marketing, Sales, Verwaltung
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.electronic-arts.de, 0221 / 975 820
Adresse: Im Zollhafen 15-17, 50678 Köln

EA Phenomic
Art des Praktikums: Vollzeit
Bereich: 2D/3D-Grafik, 2D-Grafik, 3D-Grafik, Programmierung, IT-Administration, Qualitätssicherung
Mindestdauer: 4 Monate
Ansprechpartner: Volker Pinsdorf, www.phenomic.de, 06132-783 50
Adresse: Binger Straße 38, 55218 Ingelheim

Egosoft
Art des Praktikums: Studienbegleitend oder normal
Bereich: 3D-Grafik, Design, Programmierung
Dauer: Mindestens 6 Monate
Ansprechpartner: Bernd Lehahn, bernd@egosoft.com
Adresse: Heidestraße 4, 52146 Würselen

Independent Arts Software
Art des Praktikums: Nach Absprache
Bereich: 3D-Grafik und Animation, vorzugsweise mit Softimage XSI, 2D-Grafik und Animation für GBA-Projekte, Tool, 3D-Programmierung unter C++
Mindestdauer: 3 Monate, besser 6 Monate
Ansprechpartner: Holger Kuchling, ceo@independent-arts-software.de, 02381 / 688 292
Adresse: Münsterstr. 5, 59065 Hamm

Related Designs
Art des Praktikums: Studienbegleitend, Vollzeit, Praktikumssemester
Bereich: Programmierung, Grafik, Game Design
Mindestdauer: 6 Monate
Ansprechpartner: Svenja Büll, jobs@related-designs.de
Adresse: Römerpassage 1, 55116 Mainz

RTL Games
Art des Praktikums: Vollzeit
Bereich: Akquise & Produktion
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Schoeller, dieter.schoeller@rtl.de
Adresse: Am Coloneum 1, 50829 Köln

PLZ-BEREICH 60000

Cooee GmbH
Art des Praktikums: Nach Absprache
Bereich: QA, Grafik, Webprogrammierung
Mindestdauer: 3 Monate
Ansprechpartner: Boris Karklins, jobs@clubcooee.com, 06301 / 703 290
Adresse: Opelstr. 10, 67661 Kaiserslautern

Deck 13
Art des Praktikums: Nach Absprache
Bereich: Character Modeler/Animator, Grafik/3D-Design, Macromedia-Flash-Design
Mindestdauer: 6 Monate
Ansprechpartner: Sarah Wrensch, jobs@deck13.de
Adresse: Baselerstraße 46, 60329 Frankfurt

Dutyfarm GmbH
Art des Praktikums: Studienbegleitend, Praxis-Semester
Bereich: Programmierung (Flash Actionscript, PHP, JSP, Datenbankprogrammierung), Grafik & Gamedesign (2D/3D), Produktmanagement, PR, QA
Mindestdauer: 6 Monate
Ansprechpartner: Daniel Schmitt, jobs@dutyfarm.com
Adresse: Oranienstraße 6, 10997 Berlin

extra toxic
Art des Praktikums: Studienbegleitend, nach Absprache
Bereich: Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Ivica Aracic, jobs@extratoxic.com, 069/ 408 971 43
Adresse: Hanauer Landstr. 196a, 60314 Frankfurt

Keyfactor
Art des Praktikums: Nach Absprache
Bereich: Produktionsassistenz, Marketing & PR, Asset & Content Management
Mindestdauer: 6 Monate
Ansprechpartner: Frank Fay, jobs@key-factor.de
Adresse: Rudolf-Diesel-Str. 20a, 65760 Eschborn/Ts.

MYTIVI
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Entwicklung und Konzeption TV-Games, IPTV
Mindestdauer: 6 Monate
Ansprechpartner: Daniel Schmitt, jobs@mytivi.com
Adresse: Oranienstraße 6, 10997 Berlin

World Cyber Games
Art des Praktikums: Nach Absprache
Bereich: PR
Mindestdauer: 3 Monate
Ansprechpartner: Thomas von Treichel, tvt@ac-entertainment.com, 06152 / 985 482
Adresse: Im Neugrund 6-13, 64521 Groß Gerau

PLZ-BEREICH 70000

Feenix
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Programmierung (Java, PHP)
Mindestdauer: 6 Monate
Ansprechpartner: Uwe Spitzmüller, uwe.spitzmueller@feenix.de
Adresse: Durmersheimerstr. 55, 76185 Karlsruhe

Filmakademie Baden-Württemberg
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Character Design, Engine Entwicklung, Shader Development, Cell BE Programmierung, Maya Api Tool Entwicklung
Mindestdauer: 6 Monate
Ansprechpartner: Volker Helzle, research@filmakademie.de
Adresse: Mathildenstraße 20, 71638 Ludwigsburg

Gameforge
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Grafik-Design, Web-Design, Programmierung PHP, 3D-Modelling, Game Design, Community Management
Mindestdauer: 3 bis 6 Monate
Ansprechpartner: Anett Graf, jobs@gameforge.de
Adresse: Albert-Nestler-Straße 8, 76131 Karlsruhe

Korion Simulation
Art des Praktikums: studienbegleitend, Praxis-Semester, Diplom, Bachelor
Bereich: Flash-Entwicklung, AS3-Programmierung, Webdesign, 3ds Max
Mindestdauer: 3 Monate
Ansprechpartner: Oliver Korn, oliver.korn@korion.de
Adresse: Mömpelgardstraße 16, 71640 Ludwigsburg

kr3m. media
Art des Praktikums: Vollzeit
Bereich: Programmierung Flash, Pflege von PHP-, HTML- und XML-Seiten, Illustration, Webdesign (Flash, Photoshop), Projektmanagement
Mindestdauer: mindestens 3 Monate
Ansprechpartner: Franziska Reichert, franziska.reichert@kr3m.com
Adresse: Kaiserstraße 158, 76133 Karlsruhe

SEW-EURODRIVE
Art des Praktikums: Vollzeit
Bereich: Managed DirectX Programmierung
Mindestdauer: 3 Monate
Ansprechpartner: Theres Petrasch, theres.petrasch@sew-eurodrive.de, 07251 / 751 908
Adresse: Ernst-Blickle-Str. 42, 76646 Bruchsal

Trinigy
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Softwareentwicklung im 3D-Game-Engine-Umfeld
Mindestdauer: 3 Monate
Ansprechpartner: Felix Röken, felix.roeken@trinigy.de, 07121 / 986 993
Adresse: INKA-Businesspark, Arbachtalstr. 6, 72800 Eningen

PLZ-BEREICH 80000

Airgamer
Art des Praktikums: Vollzeit, gerne auch Praxissemester
Bereich: Web Development, Redaktion
Mindestdauer: 4 bis 6 Monate
Ansprechpartner: Katja Knecht, jobs@airgamer.de, 089 / 461 382-0
Adresse: Goethestraße 10, 80336 München

B-Alive
Art des Praktikums: Nach Absprache
Bereich: Datenmanagement, Gamedesign und assistierende Projektleitung, 3D-Grafik, Software-Entwicklung C++
Mindestdauer: 6 Monate
Ansprechpartner: Arndt Schlichtig, jobs@b-alive.de
Adresse: Biberacher Straße 38, 88444 Ummendorf

Between Us!
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: PR und Marketing
Mindestdauer: 2 Monate
Ansprechpartner: Wolfram von Eichborn, veichborn@betweenus.de, 089 / 454 578 16
Adresse: Leonhard-Strell-Str. 11, 85540 Haar bei München

Disney Interactive
Art des Praktikums: Vollzeit
Bereich: Marketing
Mindestdauer: 6 Monate
Ansprechpartner: Christina Ettelbrück, christina.ettelbrueck@disney.com
Adresse: Kronstadter Str. 9, 81677 München

Marchsreiter Communications
Art des Praktikums: Vollzeit
Bereich: PR
Mindestdauer: 3 Monate
Ansprechpartner: Dieter Marchsreiter, dm@marchsreiter.com
Adresse: Guldeinstr. 41a, 80339 München

Realmforge Studios
Art des Praktikums: Nach Absprache
Bereich: Programmierung
Mindestdauer: 6 Monate
Ansprechpartner: Christian Wolfertstetter, christian.wolfertstetter@realmforgestudios.com
Adresse: Herzog-Heinrich-Straße 31, 80336 München

Remote Control Productions
Art des Praktikums: Vollzeit
Bereich: Game Design, Produktmanagement, Projektmanagement, Producing
Mindestdauer: 6 Monate
Ansprechpartner: Johannes Lesser, jobs@r-control.de
Adresse: Sendlinger-Tor-Platz 6, 80336 München

SevenOne Intermedia
Art des Praktikums: Vollzeit
Bereich: Marketing, Produkt Management, Game-Producing
Mindestdauer: 6 Monate
Ansprechpartner: David Mohr, david.mohr@sevenoneintermedia.de
Adresse: Medienallee 6, 85774 Unterföhring

Take 2 Interactive
Art des Praktikums: Vollzeit
Bereich: Marketing, Produktmanagement
Mindestdauer: 6 Monate
Ansprechpartner: Jochen Till, jochen.till@take2.de, 089 / 278 221 00
Adresse: Agnesstr. 14, 80798 München

Vivendi Games Deutschland
Art des Praktikums: Vollzeit
Bereich: Marketing
Mindestdauer: 6 Monate
Ansprechpartner: Steffen Euringer, internzen@vgames.net, 089 / 999 817 0
Adresse: Fraunhoferstr. 7, 85737 Ismaning

PLZ-BEREICH 90000

CipSoft GmbH
Art des Praktikums: Vollzeit
Bereich: Programmierung (C++, Java, HTML, XML, Linux, ggf. Grafikprogrammierung)
Mindestdauer: 3 Monate
Ansprechpartner: Bianca Patuzzi, jobs@cipsoft.com
Adresse: Gabelsbergerstraße 11, 93047 Regensburg

Feenix
Art des Praktikums: Vollzeit, Praxis-Semester
Bereich: Programmierung (Java, PHP)
Mindestdauer: 6 Monate
Ansprechpartner: Uwe Spitzmüller, uwe.spitzmueller@feenix.de
Adresse: Durmersheimerstr. 55, 76185 Karlsruhe

HandyGames
Art des Praktikums: Studienbegleitend, Vollzeit
Bereich: Grafik, Programmierung, Game- & Leveldesign
Mindestdauer: 6 Monate
Ansprechpartner: Markus Kassulke, jobs@handy-games.com, 09334 / 975 70
Adresse: i_Park Klingholz 13, 97232 Giebelstadt

qforge
Art des Praktikums: Vollzeit
Bereich: Konzeption, Programmierung (Flash/Actionscript), Systemintegration/Servermanagement (Debian Linux)
Mindestdauer: 3 Monate
Ansprechpartner: Martin Schiele, M.Schiele@qforge.de
Adresse: Südwestpark 37-41, 90449 Nürnberg

Upjers
Art des Praktikums: Vollzeit, nach Absprache
Bereich: 3D-Grafik, Programmierung, PR
Mindestdauer: 6 Wochen
Ansprechpartner: Marika Schmitt, m.schmitt@upjers.com
Adresse: Seehofstraße 46, 96117 Memmelsdorf

A-1000

Sproing
Art des Praktikums: Vollzeit, nach Absprache
Bereich: Quality Assurance, Production
Mindestdauer: 5 Monate
Ansprechpartner: Johanna Schober, jobs@sproing.com, +43 / (0)1 / 604 302 838
Adresse: Fernkorngasse 10, 1100 Wien, Österreich

IDG COMMUNICATIONS MEDIA AG

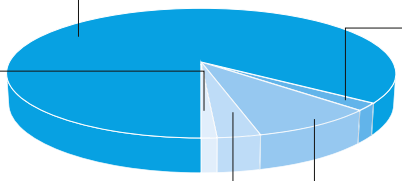
CIO ChannelPartner COMPUTERWOCHE DigitalWorld GAMEPRO GameStar Macwelt PCWELT TECCHANNEL IT EXPERTS INSIDE

Lyonel-Feininger-Straße 26 · 80807 München · Tel. 0 89 / 3 60 86-0 · Fax 0 89 / 3 60 86-118 · www.idg.de

Ausgewählte Umfragen
Dataflow

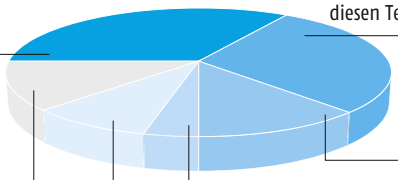
Die Redaktionen von GamePro und GameStar machen monatlich zahlreiche Umfragen unter Heftkäufern und Homepage-Usern. Eine kleine Auswahl finden Sie auf dieser Seite.

Wenn Sie am PC spielen, sind Sie dann auch mit dem Internet verbunden?


Ja, mein PC ist permanent im Internet (Flatrate). 85%
Fast immer. Ich muss mich allerdings jeweils mit dem Internet verbinden (Internet by Call, Dial-up). 2%
Keine der Antworten trifft zu. 1%
Ich bin während des Spielens nur online, wenn es absolut notwendig ist (z.B. Online-Rollenspiele, Multiplayer, Chat). 9%
Nein, mein Spielerechner hat keinen Internetanschluss. 3%

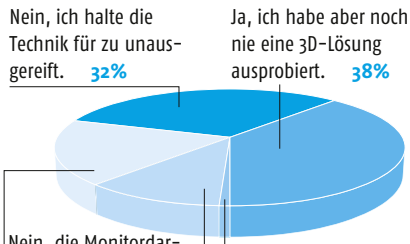
Quelle: Umfrage auf gamestar.de, 10.212 Teilnehmer

Die PC-Version von Call of Duty: Modern Warfare 2 kostet 60 Euro, 10 Euro mehr als sonst. Halten Sie den Preisanstieg für gerechtfertigt?


Der Preis ist zu hoch, ich warte auf die Budget-Version. 34%
Ich finde die Preiserhöhung unverschämt und werde diesen Teil der Serie auslassen. 29%
Ich habe generell kein Interesse an Call of Duty-Spielen. 12%
Ich finde den Preis übertrieben, werde es aber dennoch kaufen. 12%
Ja, die Qualität der Kampagne und der umfangreiche Multiplayer rechtfertigen den Preis. 4%
Nur wenn auch die Solo-Spielzeit im Vergleich zum Vorgänger ansteigt. 9%

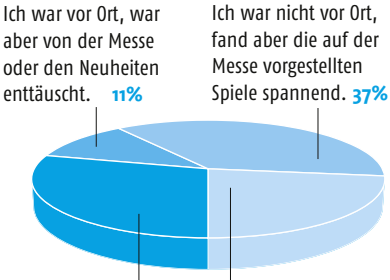
Quelle: Umfrage auf gamestar.de, 6.096 Teilnehmer

Interessiert Sie eine dreidimensionale Darstellung bei Spielen?


Nein, ich halte die Technik für zu unausgereift. 32%
Ja, ich habe aber noch nie eine 3D-Lösung ausprobiert. 38%
Nein, die Monitordarstellung sollte flach bleiben. 18%
Ja, ich besitze auch bereits eine Shutter-Brille und einen passenden Monitor. 1%
Ja, ich habe es auch bereits mit einer Rot-Grün-Brille ausprobiert. 11%

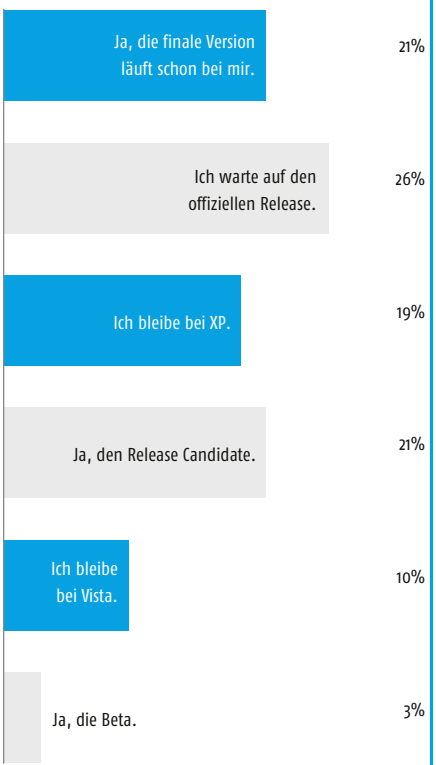
Quelle: Redaktionelle GameStar-Umfrage

Die gamescom, Europas wichtigste Spielemesse, hat ihre Pforten geschlossen. Wie lautet Ihr Messefazit?


Ich war vor Ort, war aber von der Messe oder den Neuheiten enttäuscht. 11%
Ich war nicht vor Ort, fand aber die auf der Messe vorgestellten Spiele spannend. 37%
Ich war vor Ort und fand es klasse. 29%
Die gamescom hat mich nicht sonderlich interessiert. 23%

Quelle: Umfrage auf gamepro.de, 1.145 Teilnehmer

Haben Sie Windows 7 schon ausprobiert?


Ja, die finale Version läuft schon bei mir. 21%
Ich warte auf den offiziellen Release. 26%
Ich bleibe bei XP. 19%
Ja, den Release Candidate. 21%
Ich bleibe bei Vista. 10%
Ja, die Beta. 3%

Quelle: Umfrage auf gamestar.de, 8.863 Teilnehmer

Der schlafende Riese? Brasilien – eine Marktbetrachtung

Auf neuen Märkten wird stets mit anderen Regeln gespielt. Andreas Stock gewährt uns Einblick in die wachsende brasilianische Spiele-Industrie und erklärt, warum Onlinespiele nicht ohne Rubbellose funktionieren.

Andreas Stock

ist Unternehmensberater und geschäftsführender Gesellschafter von Ganz & Stock.

Andreas Stock beschäftigt sich seit mehr als 15 Jahren mit der Vermarktung von Video- und Computerspielen. In seiner Laufbahn war er unter anderem bei Sega, Intel, der Games Academy und bei Activision im Marketing und als Geschäftsführer tätig. Neben seiner Firma Ganz & Stock, die sich auf Personalberatung für die Games-Branche spezialisiert hat, arbeitet Andreas Stock auch als Dozent an der Games Academy und ist Jurymitglied des Deutschen Computerspielpreises. Er lebt in Regensburg und zeitweise in Sao Paulo.
a_stock@t-online.de

Zur Vorgeschichte: Als ich vor sechs Jahren das erste Mal nach Brasilien reiste, hatte ich noch gar keine Vorstellung von diesem Land. Ich wollte eigentlich nur die Familie meiner Frau kennenlernen und dort Urlaub machen. Was ich nicht bedacht hatte: Die Stadt São Paulo ist eine Megacity mit über 20 Millionen Einwohnern. Es gibt dort mehr Hubschrauber als in New York und weitaus mehr (!) Hochhäuser. Der Verkehr ist mit dem zunehmenden Wohlstand der Bevölkerung immer schlimmer geworden, und auch die eine Mini-U-Bahn im Stadtzentrum São Paulos vermag die unzumutbare Situation nicht zu verbessern.

Dabei hat São Paulo als Stadt mit einem vermeintlichen Schwellenland eigentlich nicht viel zu tun. Als Deutscher bemerkt man sofort das fehlende Umweltbewusstsein, die Favelas (Armenviertel) am Stadtrand und die Tatsache, dass es so gut wie keine alten Gebäude gibt.

Vertrieb

Zu der Zeit meines Besuchs in Brasilien war ich noch als Geschäftsführer für Activision in Deutschland tätig. Und wie es im Urlaub so ist, habe ich bei jeder Shopping-Attacke meiner Frau »Store Checks« gemacht, um zu sehen, wie Computerspiele in Brasilien verkauft werden. Wie in Deutschland findet man sie bei Spezialhändlern, im Internet, in Kaufhäusern und auch in Supermärkten. Allerdings wird man weder auf einen Media Markt noch auf einen Saturn stoßen. Zwar gibt es hier und da ein paar Elektronikfachmärkte, die aber nur wenig Einfluss auf den Markt haben. Dafür werden PC-Spiele an Kiosken verstärkt in Verbindung mit Zeitschriften angeboten. Und, ganz wichtig: Raubkopien bekommt man buchstäblich an jeder Ecke. Allein in unserer Straße gibt es rund zehn »fliegende Händler«, die ihre Waren feilbieten. Die Polizei kümmert sich nicht um das Problem, da sie weit wichtigere Kriminalitätsdelikte in der Stadt aufklären muss.

Das iPhone-Musikspiel **Drums Challenge** ist derzeit Brasiliens erfolgreichster Spiele-Export.

Entwicklerstudios

Laut einer Abragames-Studie gibt es in der brasilianischen Spiele-Industrie 42 Studios und circa 600 Beschäftigte. Abragames, gegründet 2004, ist der Entwicklerverband und damit das brasilianische Pendant zum deutschen G.A.M.E. Er ist nicht nur sehr aktiv, sondern auch erfolgreich. So wurden 2009 im Rahmen des Programms »BRGAMES« vom Staat erstmalig Fördergelder für Prototypen bzw. Spielideen gezahlt. Dabei werden 70.000 bis 140.000 brasilianische Real (R$) vergeben (umgerechnet 27.000 bis 54.000 Euro). Das ist zwar an strenge Voraussetzungen geknüpft, aber immerhin ein Anfang.

In Brasilien organisiert sich die Branche auch in verschiedenen IGDA-Chapters. Publisher sind in dem Branchenverband allerdings nur sehr rudimentär vertreten. Ubisoft baut derzeit mehrere Studios auf, EA Mobile unterhält in São Paulo ein Mobile-Games-Studio.

Gutes Personal für die Entwicklung lässt sich in Brasilien wesentlich schwerer finden als bei uns, da die Industrie noch sehr jung ist. Zusätzlich fließt Risikokapital aufgrund des florierenden, exportorientierten Rohstoff- und Lebensmittelhandels nicht oder nur sehr gering in neue Technologien. Bleiben also nur die Banken mit ihren horrenden Wucherzinsen oder die Finanzierung aus dem Cashflow.

Das Team der **MusiGames Studios** gemeinsam mit ihrem erfolgreichen Erstlingswerk.

Einige Studios haben sich auf DS und Wii konzentriert, mit dem iPhone tun sich aktuell nun auch für kleinere Studios Chancen auf. Auf der GDC in San Francisco hat Konami in einer Präsentation folgende Nutzerzahlen zu diesem Thema präsentiert:

- Mobilfunkverträge in Südamerika: 411 Millionen
- davon allein in Brasilien: 160 Millionen
- zum Vergleich Nordamerika: 278 Millionen

Horrende Preise

Die Kosten für Hard- und Software übersteigen in Brasilien teils deutlich die europäischen Preise. Hier einige Zahlenbeispiele, die ich bei Onlinehändlern gefunden habe:

Hardwarepreise (Wechselkurs 1 € = 2,6 R$)
Microsoft Xbox 360: 1.299 Real (400 Euro)
In Deutschland: 239 Euro

Sony PlayStation 3 (80 GB): 1.699 Real (653 Euro)
In Deutschland: 299 Euro, inklusive Spiel

Nintendo Wii: 849 Real (326 Euro)
In Deutschland: 199 Euro

Software-Preise
Xbox- und PlayStation-3-Titel: 200 Real (76 Euro)
Nintendo Wii: 180 Real (69 Euro)
PC-Spiele: 150 Real (57 Euro)

Bei diesen Preisen muss zusätzlich bedacht werden, dass das Durchschnittseinkommen eines Brasilianers weit unter dem eines Deutschen liegt. Die knackigen Preise kommen vor allem durch viel zu hohe Importzölle zustande. Der Grund dafür ist mir jedoch unverständlich. Es gibt keine Firma in Brasilien, die in einer Konkurrenzsituation eine Konsole wie die PlayStation herstellen könnte und dementsprechend geschützt werden müsste.

Trotz der Standortschwierigkeiten wird der brasilianische Markt teils sehr positiv bewertet. So beziffert Bill van Zyll, Geschäftsführer von Nintendo in Lateinamerika, seinen Wert auf rund zwei Milliarden Dollar – Tendenz steigend. Allerdings existieren zu dieser Einschätzung keine überprüfbaren Zahlen.

Zensur auch in Brasilien?

Restriktionen für Videospiele sind bei uns in Deutschland ein häufig diskutiertes Thema. Ähnlich verhält es sich in Brasilien. Viele gewalthaltige Titel dürfen nicht im Handel verkauft werden. Doom, Mortal Combat, Counterstrike und sogar Everquest sind nur einige Beispiele für die harte Indizierungspolitik der Regierung.

Raubkopien

Raubkopien werden auf offener Straße auf kleinen Pappkartons verkauft. Es gibt eine große Auswahl an Spielen, Film-DVDs oder Anwenderprogrammen. Die Behörden gehen nur gegen gewerbsmäßig organisierte, meist von eingewanderten Asiaten geführte Firmen vor, die Produkte von Bekleidung bis hin zu Elektronikartikeln unversteuert ins Land schmuggeln. Während meiner Zeit in Brasilien wurden zweimal im Jahr in der Rua Augusta (eine Hauptstraße in São Paulo) 90 Prozent der Shops in einem kleinen Einkaufscenter dichtgemacht. Dort wurde Ware aus Asien verkauft, die nicht versteuert war und auf verschiedensten Wegen ins Land geschmuggelt wurde. Gegen einzelne Personen auf der Straße wird hingegen nicht vorgegangen.

Onlinespiele

In Brasilien leben mehr als 160 Millionen Menschen. Landessprache ist brasilianisches Portugiesisch. Da das Portugiesisch in Portugal und

Deutsch-brasilianische Koproduktion: Im Auftrag von rondomedia hat Calibre Games das DS-Spiel **Hexe Lilli entdeckt Europa** entwickelt.

Die Website des **IGDA-Chapters** in São Paulo, in dem auch die brasilianischen Entwicklerstudios von Electronic Arts und Ubisoft vertreten sind.

Im brasilianischen Branchenverband **Abragames** sind immerhin 35 Entwicklerfirmen vertreten.

Brasilien unterschiedlich ist, sollte man als Spiele-Entwickler unbedingt Sprachtrainer aus Brasilien hinzuziehen. Die Portugiesen haben als ehemalige Kolonialmacht einen extrem schlechten Ruf im Land. Dementsprechend achten die Brasilianer auf ihre eigene Aussprache und ihren eigenen Wortschatz.

Besonders deutsche Browsergames-Hersteller haben die Brasilianer bereits als Zielgruppe im Visier. Bei vielen Anbietern rangiert Brasilien unter den Top-5-Kunden, bei einigen sogar unter den Top 2 an aktiven Usern. Das liegt sicherlich daran, dass die Brasilianer generell ein sehr verspieltes Volk sind. Man denke nur an den Nationalsport Fußball und die Seleção.

Vor allem für den »Free to play«-Bereich ist Brasilien prädestiniert. Es gibt unzählige Internetcafés, die einer breiten Bevölkerungsschicht einen Onlinezugang ermöglichen. Auch an Universitäten und in Unternehmen nutzen die Brasilianer gern die Möglichkeit, online zu spielen. Denn wie oben erwähnt kann sich kaum jemand Konsolenspiele leisten. Eine Besonderheit habe ich im Buchhandel und an den Zeitungskiosken (die es massenhaft in São Paulo gibt) gesehen. Dort werden Scratchcards (Rubbelkarten) mit Spielguthaben sowie Lösungsbücher und Hefte für Onlinespiele verkauft. Die Internetcafés sind teilweise mit Postern oder Handzetteln der Onlinespiele ausgestattet. In einigen Läden ist sogar Verkaufspersonal der Spielehersteller direkt vor Ort, um Spielguthaben zu verkaufen – und das scheint zu funktionieren. Denn die Menschen sind gern bereit, ein paar Euro für gute Unterhaltung zu zahlen. Außerdem kann man das Spiel erst ausprobieren und wird so nicht übers Ohr gehauen. Die Brasilianer sind von Natur aus sehr spielfreudig, bleiben aber auch ihren Lieblingstiteln treu. Mein Übersetzer versicherte mir sogar, dass alle seine Schüler Onlinegames spielten.

Einziger Wermutstropfen bei manchen Anbietern von Browsergames: Die Conversion Rate vom aktiven zum zahlenden User ist nicht optimal. Das kann von Fall zu Fall sicher am Spiel liegen. Für mich gibt es aber zwei Hauptgründe: Einerseits kann man über die großen Portale in Brasilien zwar Masse, aber keine Klasse generieren. Hier wäre eine detaillierte Marktbearbeitung notwendig, um kein Potenzial zu verschenken. Außerdem ist das Payment ein Problem: Die wenigsten jungen Leute haben ein Bankkonto. Dementsprechend schwer fällt die Bezahlung im Netz. Denn eine Kreditkarte ohne Bankkonto gibt es logischerweise nicht. SMS Payment nutzt fast niemandem. Als Lösung bieten sich mehrere Möglichkeiten an. Das Boleto Bancário etwa, bei dem man in die Bank laufen muss (und in der Regel eine halbe Stunde wartet, bis man an der Reihe ist!), um Geld einzuzahlen. Die weitaus bessere Möglichkeit ist der oben erwähnte Verkauf von Rubbelkarten am Kiosk oder in Internetcafés. Dies funktioniert aber natürlich nur, wenn man die lokalen Gegebenheiten kennt – für ausländische Spieleheresteller ein nicht zu unterschätzendes Problem.

Erfolgsgeschichte iPhone

Als Beispiel für eine erfolgreiche Studio-Neugründung ist vor allem das MusiGames Studio in Recife zu nennen. 2008 im »Porto Digital« (der größte Innovation Park Lateinamerikas) gegründet, hat sich das Unternehmen auf die Entwicklung von iPhone-Spielen konzentriert. In der Music-Games-Kategorie hat der erste Titel Drums Challenge schon voll in den Top 5 eingeschlagen. Die musikverliebten Brasilianer haben somit einen Grundstein für weitere Titel wie Throat Detonator gelegt. Wie der Titel schon andeutet, muss man in diesem Spiel die Gegner mit seiner Stimme besiegen.

Die Brazilian Innovation Agency (FINEP) hat die Firma unterstützt, inzwischen arbeiten 23 Mitarbeiter aus dem Audio-Bereich mit den Entwicklern zusammen. Fünf weitere Titel sind derzeit in Planung.

Andreas Stock

3 Monate GameStar testen

nur 14,90 Euro!

Über 20% sparen!

GameStar XL:
2 DVDs
2 Vollversionen
Extra-Abocover

+ GRATIS: Raptor Gaming Mousepad P4!

- speziell beschichtetes und hochwertiges schwarzes Stoffmaterial
- Reibungswiderstand speziell abgestimmt auf Low-Sense-Spieler
- extrem gute Gleiteigenschaft
- rutschfeste Unterseite
- mühelose, pixelgenaue Cursor-Steuerung
- großzügige Auflagefläche (400 x 300 mm)
- kompatibel zu allen gängigen Maustypen
- lange Lebensdauer, ergonomisches Design
- geringe Höhe von nur 2 mm

Die Online-Bestellung geht am schnellsten:

www.gamestar.de/shopcode

Shop-Code 111 Shopcode merken für die Online-Bestellung

Bestellhotline Mo-Fr 9 - 17 Uhr: **0711 / 72 52 - 275** Fax: **0711 / 72 52 - 377** E-Mail: **shop@gamestar.de**

GameStar erscheint im Verlag IDG Entertainment Media GmbH, Lyonel-Feininger-Str. 26, 80807 München, Handelsregister München, HRB 116413, Geschäftsführer: York von Heimburg. Die Kundenbetreuung erfolgt durch den GameStar Kundenservice, ZENIT Pressevertrieb GmbH, Postfach 810580, 70522 Stuttgart, Geschäftsführer: Peter Staudenmaier, Tel. 0711-7252275, eMail: shop@gamestar.de.

WWW
makinggames.de
- Online-Register mit Suchfunktion und detailierten Firmeninfos

Helfen Sie uns!

Sie haben einen Fehler entdeckt oder vermissen Ihre Firma? Dann schreiben Sie uns: **info@makinggames.de, Betreff: Firmenregister**

Die Branche im Überblick
Firmenregister

Vergessen Sie die Auskunft! Hier finden Sie sämtliche uns bekannten Publisher, Entwickler, Dienstleister, Ausbilder und Verbände der deutschsprachigen Spiele-Industrie – von Acony bis Zuxxez.

PUBLISHER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Activision Blizzard Deutschland GmbH	Fraunhoferstraße 7	85737 Ismaning	089 / 9998170	info@activision.de
Aerosoft	Lindberghring 12	33142 Büren	02955 / 760310	info@aerosoft.de
Application Systems Heidelberg	Pleikartsförsterhof 4/1	69124 Heidelberg	06221 / 300002	kontakt@application-systems.de
Astrum Online Entertainment	Rothenbaumchaussee 7	20148 Hamburg	040 / 4143420	info@astrumonline.de
bad brain entertainment	Benzstraße 1b	51381 Leverkusen	02171 / 582667	team@bad-brain.com
BHV	Novesiastraße 60	41564 Kaarst-Büttgen	02131 / 76501	info@bhv.de
BIGPOINT GmbH	Christoph-Probst-Weg 3	20251 Hamburg	040 / 881413501	info@bigpoint.net
Capcom – CEG Interactive Entertainment	Barmbeker Straße 4b	22303 Hamburg	040 / 6965620	info@capcomeuro.de
CDV	Werner-von-Siemens-Str. 9	76646 Bruchsal	07251 / 50505601	mail@cdv.de
Codemasters	Millerntorplatz 1	20359 Hamburg	0319 / 747202	online@codemasters.de
Comport Interact!ve	Eberleinstraße 46	65195 Wiesbaden	0611 / 53169783	info@comport-interactive.com
Dartmoorsoft	Postfach 6161	32647 Lemgo	05261 / 660731	info@dartmoorsoft.com
Disney Interactive Studios	Kronstadter Straße 9	81677 München	089 / 99340191	DIS.Germany.Marketing@disney.com
dtp entertainment AG	Goldbekplatz 3-5	22303 Hamburg	040 / 6699100	info@dtp-ag.com
dtp young entertainment GmbH & Co. KG	Goldbekplatz 3-5	22303 Hamburg	040 / 6699100	info@dtp-young.com
Eidos / Sci	Große Elbstraße 145d	22767 Hamburg	040 / 30633400	info@eidos.de
Electronic Arts	Im Zollhafen 15-17	50678 Köln0221 / 975820	de-login@ea.com	
Empire	Emmy-Noether-Straße 9	76131 Karlsruhe	0721 / 6105300	stephanie.mueller@empire.co.uk
Flashpoint	Biedenkamp 5h	21509 Glinde/Hamburg	040 / 710060	info@flashpoint.de
Frogster Interactive Pictures	Hardenbergstraße 9A	10623 Berlin	030 / 2847010	info@frogster-ip.de
gamigo	Behringstraße 16b	22765 Hamburg	040 / 226305250	gamigoag@gamigo-mail.de
Gameforge	Albert-Nestler-Straße 8	76131 Karlsruhe	0721 / 3548080	info@gameforge.de
Halycon Media	Alte Dorfstraße 20a	23860 Klein Wesenberg	04533 / 6109-0	info@halycon.de
Headup Games	Nordstraße 104	52353 Düren	02421/ 4868703	zolna@headupgames.com
Jowood	Technologiepark 4a	A-8786 Rottenmann	+43 / 3614 / 29660	office@jowood.com
Kalypso Media	Prinz-Carl-Anlage 36	67547 Worms	06241 / 502240	info@kalypsomedia.com
Koch Media	Lochhamer Straße 9	82152 Planegg/München	089 / 24245120	office@kochmedia.com
Konami	Berner Straße 103-105	60437 Frankfurt am Main	069 / 9855730	support@konami.de
Lomax Software	Postfach 1120	48482 Neuenkirchen	02551 / 919372	lomaxsoft@aol.com
Microsoft	Konrad-Zuse-Straße 1	85716 Unterschleißheim	089 / 31760	service@service.microsoft.de
Millform	Ringstraße 110	CH-4106 Therwil	+41 / 61 / 7220712	werner.sala@millform.ch
Morphicon	Lindwurmstraße 88	80337 Munich	089 / 45235460	info@morphicon.de
MultiMediaManufaktur	Ludwigstraße 33-37	60327 Frankfurt am Main	069 / 974600	info@multimediamanufaktur.com
Namco Bandai Partners	Carl-Benz-Straße 21	60386 Frankfurt am Main	069 / 24449100	
NBG	Brunnfeld 2-6	93133 Burglengenfeld	09471 / 70170	info@nbg-online.de
Nintendo	Nintendo Center	63760 Grossostheim	06026 / 9500	info@nintendo.de
Nokia	Heltorfer Straße 1	40472 Düsseldorf	0190 / 772222	
Novitas	Frankfurter Straße 2	38122 Braunschweig	0531 / 215340	kontakt@peppergames.de
Phenomedia	Friedrich-Ebert-Straße 14	44866 Bochum	02327 / 9970	Info@phenomedia.com
Rondomedia	Limitenstraße 64-78	41236 Mönchengladbach	02166 / 618660	info@rondomedia.de
RTL Games	Am Coloneum 1	50829 Köln0221 / 7800	team@rtl-games.de	
S.A.D.	Rötelbachstraße 91	89079 Ulm 07305 / 96290	office@s-a-d.de	
Sega	Rablstraße 24	81669 München	089 / 22848000	info@sega.de
SevenOne Intermedia	Medienallee 6	85774 Unterföhring	089 / 95078900	kontakt@sevenoneintermedia.de
Sony Computer Entertainment Deutschland	Frankfurter Straße 233	63263 Neu-Isenburg	06102 / 4330	scee_presscentre@scee.net
Soulfood	Agathe-Lasch-Weg 2	22605 Hamburg	040 / 8541960	nfo@soulfood-entertainment.com
Take 2	Agnesstraße 14	80798 München	089 / 278220	info@take2.de
TDK / Playlogic	Halskestraße 38	40880 Ratingen	01805 / 83542637	tdkgames@maxupport.de
The Games Company	Ehrenbergstraße 19	10245 Berlin	030 / 526853000	info@the-games-company.com
THQ	Kimplerstraße 278	47807 Krefeld	02151 / 41890	thq@thq.de
Tivola Electronic Publishing	Münzstraße 19	10178 Berlin	030 / 5363580	mail@tivola.de
Ubisoft	Adlerstraße 74	40211 Düsseldorf	0211 / 338000	info@ubisoft.de
Zuxxez	Rittnertstraße 36	76227 Karlsruhe	01805 / 989939	info@zuxxez.com

ENTWICKLER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
3d-io	Rüdesheimer Straße 9	65197 Wiesbaden	0611 / 50462620	info@3d-io.com
49Games	Eimsbütteler Straße 64	22769 Hamburg		kontakt@49games.de
Acony	Mönchweilerstraße 1/8	78048 Villingen-Schwenningen	07721 / 9983829	
B-Alive	Biberacher Straße 38	88444 Ummendorf		contact@b-alive.de
BigBlackBlock Gamestudio	Am Veenteich 1	46147 Oberhausen	0208 / 6101995	info@bigblackblock.com
BIGPOINT GmbH	Christoph-Probst-Weg 3	20251 Hamburg	040 / 881413501	info@bigpoint.net
binary madness	Bundesallee 55	10715 Berlin	030 / 25093746	info@binarymadness.de
Bitfield	Rigaer Straße 56	10247 Berlin	030 / 40045181	kontakt@bitfield.de
Blue Byte	Adlerstraße 74	40211 Düsseldorf	0211 / 33800500	info@bluebyte.de
BlurredVision	Glashütter Weg 80	22844 Norderstedt		info@blurredvision.de
Boiling Point Studios	Starkenburger Straße 5	14163 Berlin		fkollmann@boilingpoint-studios.com
Buntware Games	Colmarerstrasse 9	79106 Freiburg	0761 / 2046988	contact@buntware.com
BURNS Entertainment GmbH	Marienburger Straße 26	10405 Berlin	030 / 44050281	info@burns-games.com
Bright Future	Im Zollhafen 15-17	50678 Köln0221 / 78982171	info@brightfuture.de	
Bytedefenders	Eldenaer Straße 17	10247 Berlin	030 / 42019164	info@bytedefenders.com
ByteRockers	Schönhauser Allee 36	10435 Berlin	030 / 44318610	info@byterockers.de
caipirinha games	Kiefernweg 1a	24161 Altenholz	0431 / 324540	info@caipirinhagames.de
cerasus.media	Ostseestraße 109	10409 Berlin	030 / 802089990	info@cerasus.de
Chimera Entertainment	Augsburger Straße 8	80337 München	089 / 28979940	info@chimera-entertainment.com
CipSoft	Gabelsbergerstraße 11	93047 Regensburg	0941 / 6308280	kontakt@cipsoft.com
Claw Design	Ludwigstraße 8	72474 Winterlingen	07434 / 3931	info@clawdesign.com
Cooee GmbH	Opelstraße 10	67661 Kaiserslautern	06301 / 703290	info@clubcooee.com
Cranberry Production	Podbielskistraße 166	30177 Hannover	0551 / 76170000	mail@cranberry-production.com
CreaTeam Software	Sünderup 6	24943 Flensburg	0461 / 64808	
Crenetic GmbH	Hingbergstraße 99	45468 Mülheim an der Ruhr	0208 / 4442143	info@crenetic.de
Crimson Cow	Tarpen 40	22419 Hamburg	040 / 40132614	kontakt@crimsoncow.de
Crytek	Hanauer Landstraße 523	60386 Frankfurt am Main		business@crytek.de
Daedalic Entertainment	Workport Unit 2	22453 Hamburg	040 / 432612720	info@daedalic.de
Decane	Hallerstraße 5e	20146 Hamburg	040 / 42910947	info@decane.net
Deck13 Interactive	Baselerstraße 46	60329 Frankfurt	069 / 7167660	info@deck13.de
Destrax Entertainment	Merkurstraße 10	33739 Bielefeld		info@destrax.de
Dion Software	Hegelgasse 6/10	A-1010 Wien	+43 / 1 / 51260350	office@dion-software.com
dreiNET	Bergedorfer Straße 96	12623 Berlin		philip@dreinet.org
Dutyfarm	Oranienstraße 6	10997 Berlin	030 / 30368430	info@dutyfarm.com
EA Phenomic	Binger Straße 38	55218 Ingelheim	06132 / 78350	info@phenomic.net
Egosoft	Heidestraße 4	52146 Würselen		egoinfo@egosoft.com
Eku Interactive	Saalburgstraße 104	61350 Bad Homburg	06172 / 597727	office@eku.de
Elkware	Grenzweg 1	22880 Wedel	04103 / 902090	gameseurope@infospace.com
Elocom Mobile Entertainment	Bad Essener Straße 15d	49143 Bissendorf	05402 / 9855900	elo@elocom.de
eos interactive	Rigaer Straße 56/I	10247 Berlin	030 / 40045180	info@eosinteractive.de
eWave Interactive	Dennis-Gabor-Straße 2	14469 Potsdam	0331 / 7309730	info@ewave.de
exDream GmbH	Schaufelder Straße 11	30167 Hannover	0511 / 3705967	info@exdream.com
Exozet Games	Rotherstraße 20	10245 Berlin	030 / 2465600	games@exozet.com
Extra Toxic	Hanauer Landstraße 196a	60314 Frankfurt	069 / 48440040	info@extratoxic.com
Fakt Software	Rosa-Luxemburg-Straße 76	08058 Zwickau	0375 / 2000557	info@fakt-software.de
Fancy Bytes	Dießemer Straße 152	47799 Krefeld	02151 / 3259912	business@fancy-bytes.de
FDG Mobile Games	Kellerstraße 32	81667 München	089 / 32212356	info@fdgsoft.de
Freudenreich & Tollknaepper	Postfach 101 469	42014 Wuppertal	0202 / 2721048	mail@german-railroads.com
Funatics	Essener Straße 3	46047 Oberhausen	0208 / 43927060	friedmann@funatics.de
Fusionsphere Systems	Rosenau 8	86551 Aichach	0700 / 3874600	info@fusionsphere.com
GameDevelopment M. Molkenthin	Lerchenstraße 98	22767 Hamburg	040 / 46773207	gamedev@michael-molkenthin.de
Gameforge	Albert-Nestler-Straße 8	76131 Karlsruhe	0721 / 3548080	info@gameforge.de
gameus	Kufsteinerstraße 40	83064 Raubling		info@gameus.de
Gbanga	Brauerstraße 31	CH-8004 Zürich	+41 / 43 / 5366701	matthias@gbanga.com
gEasy Development	Königsbacher Straße 2	67067 Ludwigshafen	0621 / 5540660	info@geasyteam.de
GIANTS Software GmbH	Badenerstraße 621	CH-8048 Zürich		mail_us@giants-software.com
Greencode	Am Hofgarten 3	53113 Bonn		info@greencode.de
Greentube	Mariahilfer Straße 47/1/102	A-1060 Wien		office@greentube.com
HandyGames	i_Park Klingholz 13	97232 Giebelstadt	09334 / 97570	info@handy-games.com
House of Tales	Postfach 1713	57278 Neunkirchen		info@house-of-tales.com
Independent Arts Software	Münsterstraße 5/Haus 4	59065 Hamm	02381 / 688292	info@independent-arts-software.de
InnoGames	Tempowerkring 6	21079 Hamburg	040 / 78893350	info@innogames.de
Island Games	Vahrenwalder Straße 7	30165 Hannover	0511 / 35319898	kontakt@islandgames.de
Keen Games	Dreieichstraße 59	60594 Frankfurt am Main	069 / 66058838	info@keengames.com
King Art	Schlüsselkorb 17/18	28195 Bremen	0421 / 322760	contact@kingart.de
KORION Simulation	Mömpelgardstraße 16	71640 Ludwigsburg	07141 / 3898230	info@korion.de
Kritzelkratz 3000	Martinstraße 21	97070 Würzburg	0931 / 3536091	info@kritzelkratz.de
kunst-stoff GmbH	Rungestraße 22-24	10179 Berlin	030 / 23457178	info@kunst-stoff.de
La Plata Studios	Papenreye 61	22453 Hamburg	040 / 42236100	contact@laplata-studios.com
Limbic Entertainment	Heinrich-Hertz-Straße 21	63225 Langen	06103 / 960357	info@limbic-entertainment.de
Madcat Interactive Software	Prager Ring 4-12	66482 Zweibrücken	06332 / 791320	info@madcat.de
MaDe Games GbR Neetix - Wycislik	Hochstraße 34	41334 Nettetal		m.wycislik@madegames.com
marune	Wallbergstraße 16	82008 Unterhaching	089 / 665007 49	info@marune.de
Master Creating	Winterhuder Weg 82	22085 Hamburg	040 / 32901223	mail@master-creating.com
Media Seasons	Liviastraße 6	04105 Leipzig	0341 / 2156752	contact@mediaseasons.com
Mobilescope	Am Meerkamp 19	40667 Meerbusch	02132 / 1370	info@mobilescope.com
mobivention GmbH	Albin-Köbis-Straße 8	51147 Köln	02203 / 906020	info@mobivention.com
netmin games	Philipp-Reis-Straße 6	55129 Mainz	06131 / 507896	schreiber@netmin.de
Niels Bauer Games	Sickingenstraße 37	79117 Freiburg	0761 / 66947	contact@nielsbauergames.com
northworks	Stahltwiete 23	22761 Hamburg	040 / 42236100	info@northworks.de
Novatrix	Schlüsselwiesen 19	70186 Stuttgart	0711 / 3102770	info@novatrix.de
Nuclearvision	Körnerstraße 9	38102 Braunschweig	0531 / 70189840	contact@nuclearvision.de
Nurogames GmbH	Schaafenstraße 25	50676 Köln0221 / 3988040	info@nurogames.com	
OG-Soft Productions	Zirbelstraße 48	86154 Augsburg		webmaster@og-soft.de
online4ever Limited	Manitiusstraße 6	01067 Dresden	0351/ 2063876	info@online4ever.biz
Outline Development	Rappenweiher 12	57074 Siegen	0271 / 3131331	webmaster@outline-development.de

ENTWICKLER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Piranha Bytes	Ruhrallee 63	45138 Essen	0201 / 806720	info.neu@piranha-bytes.com
pixeltamer.net	Clausewitzstraße 6	10629 Berlin	0 30 / 34347690	info@pixeltamer.net
Pixon	Jaffestraße 10	21109 Hamburg	040 / 33399430	info@pixon.de
Procontis GmbH	Am Bauhof 18	64807 Dieburg	06071 / 92860	kontakt@procontis.de
Rabenauge	Kniprodestraße 18	10407 Berlin	030 / 42809100	the_contact@rabenauge.com
Radon Labs	Bouchéstrasse 12	12435 Berlin	030 /43739455	feedback@radonlabs.de
Reakktor	Arnswaldtstrasse 10	30159 Hannover	0511 / 540980	biz@reakktor.com
Realmforge Studios	Herzog-Heinrich-Straße 31	80336 München	089 / 55069134	info@realmforgestudios.com
Redfire Software	Haarmeyers Kamp 6	49586 Neuenkirchen	05465 / 9923	contact@redfiresoft.de
Related Designs	Römerpassage 1	55116 Mainz	06131 / 5544711	info@related-designs.de
Replay Studios	Barmbeker Straße 6a	22303 Hamburg	040 / 284107830	info@replaystudios.de
Rotobee	Schwedter Straße 9A	10119 Berlin	030 / 41715525	info@rotobee.com
Silent Future	Vohwinkeler Straße 51-53	42329 Wuppertal	0202 / 5143480	info@silentfuture.de
Silver Style	Am Borsigturm 12	13507 Berlin	030 / 526853000	info@the-games-company.com
Similis	Lessingstraße 6	46149 Oberhausen	0208 / 625240	info@similis.com
Sixteen Tons Entertainment	Karlstraße 3	72072 Tübingen		info@promotion-software.de
SnapDragon Games	Schulterblatt 58	20357 Hamburg	040 / 284100630	info@snapdragongames.com
Snowstep Development	Ernst-Gnoß-Straße 22	40219 Düsseldorf	0211 / 30329965	info@snowstep.com
Solimedia productions	Schlösserstraße 28	99084 Erfurt	0361 / 658550	hatmut.niemeier@solimedia.de
Spellbound	Weststraße 15	77694 Kehl am Rhein	07851 / 991660	info@spellbound.de
SpinBottle Games GmnH	Windhukweg 3	45357 Essen	0201 / 1784817 6	info@spinbottlegames.com
Spirit	Friedrichstraße 15	70174 Stuttgart	0711 / 96685510	weyrich@go4spirit.com
spotsonfire GmbH	Hongkongstraße 7	20457 Hamburg	040 / 209346330	info@spotsonfire.com
Springwald Software	Alter Eistreff 36	44789 Bochum	0700 / 777464925	info@springwald.de
Sproing	Fernkorngasse 10	A-1100 Wien	+43 / 1 / 6043028	office@sproing.com
Synetic	Auf der Haar 47	33332 Gütersloh	05241 / 4009900	info@synetic.de
Ticking Bomb Games GmbH	Adenauerallee 32	20097 Hamburg		jobs@ticking-bomb-games.de
Travian Games GmbH	Wilhelm-Wagenfeld-Straße 22	80807 München	089 / 3249150	mail@traviangames.com
Upjers	Seehofstraße 46	96117 Memmelsdorf	0951 / 3015835	mail@upjers.com
Vectorcom	Baumbachstraße 1	30163 Hannover		mail@vectorcom.de
Victory Soft Entertainment	1. Viertel 9	A-8171 St.Kathrein	+43 / 03179 / 8223	victory.soft@hotmail.at
VIS Games and Entertainment	Am Bitzenkamp 21d	50226 Frechen	02234 / 22983	info@vis-games.de
Vulcando Games	Werner-von-Siemens-Str. 6	86159 Augsburg	0821 / 4550834	info@vulcando-games.de
wetain	Burgstraße 10	41540 Dormagen	02133 / 882530	info@wetain.com
Xybris Interactive	Apfelstraße 12a	33613 Bielefeld	0521 / 9873035	info@xybris.com
Yager Development	Pfuelstraße 5	10997 Berlin	030 / 69597660	contact@yager.de
Zeroscale	Schwedter Straße 9a	10119 Berlin	030 / 405057920	info@zeroscale.com
zone2.media	Am Wasserturm 6	40668 Meerbusch	02150 / 705660	info@zone2.de

DIENSTLEISTER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
4-REAL INTERMEDIA (Lokalisation)	Ludwigstraße 32	63067 Offenbach	069 / 8090880	info@4-real.com
Akamai Technologies (Datentransfer-Technologie)	Betastraße 10B	85774 Unterföhring	089 / 94006308	sales-germany@akamai.com
Anakan (Lokalisation)	Pfuelstraße 5	10997 Berlin	030 / 531420450	ana@anakan.de
AP Lellwitz (Marktforschung)	Fichtestraße 65	63303 Dreieich	0160 / 20 143 64	info@apl-mafo.de
Artificial Technology (KI-Middleware)	Dornierstraße 2	82178 Puchheim	089 / 80076124	info@artificial-technology.com
Aruba Studios (Events)	Frintroper Straße 453	45359 Essen	0201 / 6131721	infos@aruba-studios.de
arvato (Full-Service-Dienstleister für MMOs)	An der Autobahn 18	33311 Gütersloh	05241/ 8080888	info@arvato-systems.de
Atomic-Design (Grafik)	Quedlinburger Weg 27	22455 Hamburg	040 / 41265871	info@atomic-design.de
Between Us! (PR, Marketing)	Leonhard-Strell-Straße 11	85540 Haar	089 / 45457811	info@versant.com
BiteTheBytes GbR (Middleware)	Heinrich-Von-Bibra Platz 1b	36037 Fulda	0661 / 2006899	info@bitethebytes.com
Centroid (Motion Capturing)	Schlesische Straße 27	10997 Berlin	030 / 698074 74	sven.bergmann@centroidanimation.com
Chromatrix (Mobile Gaming)	Kalkofenstraße 2	72411 Bodelshausen	07471 / 740828	info@chromatrix.com
CNT Media (3D, Animation)	Süderstraße 159a	20537 Hamburg	040 / 35085890	info@cnt-media.net
Codecult (3D-Engine)	Kemnastraße 21a	44866 Bochum	02327 / 35474	info@codecult.com
Compad Company (Merchandising)	Habsburger Straße 85	79104 Freiburg	0761 / 5853667	mz@com-pad.com
ContenTainer (Leveldesign)	Eichelkopfstraße 17	63584 Gründau	06058 / 9178150	info@contentainer.de
Creatown (Game Design, Consulting)	Diemelkoppel 5	21406 Melbeck	04134 / 900860	j.beilschmidt@creatown.de
Creature Factory	Graefestraße 33	10967 Berlin	030 / 26301330	info@creature-factory.com
definitv:ton (Audio)	Lindenthalgürtel 84	50935 Köln	0221 / 8234220	mhoechsmann@definitiv-ton.de
Die Kolonie (Art, Design)	Adelheidstraße 82	65185 Wiesbaden	0611 / 5058765	contact@diekolonie.com
Dynamedion (Sounddesign)	Barbarossaring 8	55118 Mainz	06131 / 5847895	info@dynamedion.com
Effective Media (Lokalisation)	Schwarzmannstraße 8	80798 München	089 / 38998999	info@effective-media.de
Eye Rock Media (Design, Animation)	Eichsfeldstraße 62	46147 Oberhausen	0208 / 6256800	contact@eyerock.de
fadeout (Audio Design)	Austraße 6	CH-4153 Reinach	+ 41 / 79 / 2793244	info@fadeout.ch
Falko Löffler (Autor)	Lindenstraße 8a	36355 Ilbeshausen	06643 / 918577	mail@falkoloeffler.de
Fantastic Illusions (Grafik)	Beethovenstraße 40a	48317 Drensteinfurt	02508 / 985801	info@illusions.de
Friedmann Kommunikation (Marketing / PR)	Schanzenstraße 36	51063 Köln	0221 / 5348710	info@fr-k.de
Gärtner PR (PR, Marketing)	Destouchesstraße 68	80796 München	089 / 30766854	info@gaertner-pr.de
GD Gameduell (Spiele-Plattform)	Ackerstraße 14-15	10115 Berlin	030 / 288768211	info@gameduell.de
Global Producer (Production)	Hauptstr. 172	51143 Köln	02203 / 97722450	getintouch@global-producer.com
Glare Studios (Grafik)	Mansteinstraße 18	20253 Hamburg	040 / 63948120	contact@glarestudios.de
GreenMamba-Studios (Animation)	Aloysstraße 7	48249 Dülmen	0172 / 2648745	info@greenmamba-studios.de
Hans HiScore (Audio)	Friedrich-Ebert-Anlage 11-13	60327 Frankfurt	0151 / 11616133	info@hans-hiscore.de
IGA Worldwide Ltd. (In-Game-Advertising)	Oranienburger Straße 86a	10178 Berlin	030 / 2008900	info@ingameadvertising.com
IHDE & Partner (Rechtsberatung)	Schoenhauser Allee 10-11	10119 Berlin	030 / 44318660	berlin@onlinelaw.de
IMC (Full-Service Dienstleister)	Hauptstraße 172	51143 Köln	02203 / 97722400	info@imc-online.net
Indigo Pearl GmbH & Co. KG (PR, Marketing)	Richardstraße 45	22081 Hamburg	040 / 25330266	presse@indigopearl.de
Intulo (3D, Animation)	Am Schützenholz 25	29643 Neuenkirchen	05195 / 972690	info@intulo.de
JOGO Media (In-Game-Advertising)	Graf-Adolf-Platz 15	40213 Düsseldorf	0211 / 52099491	question@jogomedia.com
Keyfactor (Consulting)	Rudolf-Diesel-Straße 20a	65760 Eschborn	06173 / 999781	contact@key-factor.de

DIENSTLEISTER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Marchsreiter Communications	Guldeinstraße 41a	80339 München	089 / 51919942	dm@marchsreiter.com
Maurice T. Anderson (Komposition und Sounddesign)	Obertorstraße 4	63454 Hanau	0176 / 26228136	contact@maurice-t-anderson.com
ME Enterprises (Consulting)	Herrenhäuser Straße 24	30938 Burgwedel	05139 / 9599915	mmoehring@meenterprises.de
MediaLancer (Animation)	Opelstraße 10	67661 Kaiserslautern	06301 / 703280	info@medialancer.com
metricminds (Grafik)	Rüsselsheimer Straße 22	60326 Frankfurt am Main	069 / 7593380	philip@metricminds.com
Michael Bhatty Entertainment (Entwicklung, Consulting)	Bismarckstraße 24	49076 Osnabrück	0541 / 1819660	michael@bhatty.de
MMO hospital (Datenbank)	Wiesenkamp 22 B	22359 Hamburg	040 / 609900	mick@MMOhospital.org
modus werbung (PR, Marketing)	Neuer Kamp 30, Eingang C	20357 Hamburg	040 / 2380560	mail@modus-werbung.de
Partnertrans (Lokalisation)	Leipziger Straße 28	40822 Mettmann	02104 / 172660	info@partnertrans.com
Pixelstorm (Grafik)	Tannenweg 25	24637 Bokhorst	04394 / 991458	bleicken@pixelstorm.de
Play + Smile Marketing (PR, Marketing)	Am Wasserturm 6	40668 Meerbusch	02150 / 705660	info@play-and-smile.de
Polyce (Grafik)	Hübnerstraße 4	10247 Berlin	0178 / 3592560	info@polyce.de
ProCom International LTD	Kisdorfer Weg 3b	24568 Kaltenkirchen	040 / 71669763	info@procom-international.com
pro-tone (Audio, Lokalisation)	Strahlenbergerstraße 125a	63067 Offenbach am Main	069 / 2475700	info@pro-tone.de
Quality Four (Qualitätssicherung)	Behlertstraße 3a	14467 Potsdam	0331 / 2008918	info@qualityfour.de
RABCAT entertainment (Grafik)	Fernkorngasse 10	A-1100 Wien	+43 / 1 / 5237425	mail@rabcat.com
remote control productions (Producing, Consulting)	Sendlinger-Tor-Platz 6	80336 München	089 / 26949648	info@r-control.de
Rode + Mathé (Rechtsberatung)	Henriettenweg 4	20259 Hamburg	040 / 43270432	info@rodemathe.de
ROESSLER PR (Marketing, PR, Consulting)	Walter-Leiske-Straße 2	60320 Frankfurt am Main	069 / 5144161	kr@roesslerpr.de
Rocketbox Studios GmbH (Grafik)	Leonhardtstraße 10	30175 Hannover	0511 / 8984384	info@rocketbox.de
SCC (Crossmedia)	Orleansstraße 5a	81669 München	089 / 33094660	info@scc-feld.de
Schanz International (Consulting)	Bgm.-Oberhettinger-Str. 2a	67146 Deidesheim	06326 / 6010	info@schanzgames.com
Schulte Riesenkampff (Rechtsberatung)	Hochstraße 49	60313 Frankfurt am Main	069 / 900266	www.schulte-lawyers.com
Selected Minds (Personalberatung)	Walther-Rathenau-Straße 16	64521 Groß-Gerau	06152 / 1871830	hello@selected-minds.de
SFK (interaktiver SpieleTV-Sender)	Alter Wandrahm 15	20457 Hamburg	040 / 348094974	info@skfernsehen.com
Spooky Cecile Media Productions (Grafik)	Chausseestraße 92	10115 Berlin	030 / 20089210	info@spooky-cecile.com
Sunny Music Musikverlag (Musikproduktion)	Hufelandstraße 13	33014 Bad Driburg	05253 / 974281	info@sunnymusic.de
swordfish pr (PR, Marketing)	Trogerstraße 17	81675 München	089 / 60031641	ilka@swordfish-pr.de
The Light Works (Grafik)	Emil-Hoffmann-Straße 27	50996 Köln	02236 / 967322	info@thelightworks.com
Tiger Team Productions (Produktion)	Bahnhofstraße 27	61184 Karben	06039 / 484525	ralf@tigerteam-productions.de
ToBringAlive (Animation, Design)	Ostmarkstraße 49	48145 Münster	0251 / 392312	angerbauer@tobringalive.com
Tom Putzki Consulting	Behrensstraße 43	44623 Herne	02323 / 9190053	tp@tom-putzki-consulting.com
Toygardens Media (Grafik, Video)	Hubertusstraße 2	30163 Hannover	0511 / 33659800	akunze@toygardens.com
Trinigy (3D-Engine)	Arbachtalstraße 6	72800 Eningen u.A.	07121 / 986993	info@trinigy.de
Turtle Entertainment (eSport)	Siegburger Straße 189	50679 Köln 0221 / 8804490	info@turtle-entertainment.de	
Two Pi Team (Community Management, Lokalisation)	Gerresheimer Straße 9	40721 Hilden	02103 / 32276	info@two-pi-team.de
VCC Perfect Pictures (Grafik, Video)	Graf-Adolf-Straße 22	40212 Düsseldorf	0211 / 828590	o_rudolph@vcc.de
Versant (Data Management)	Wiesenkamp 22b	22359 Hamburg	040 / 609900	info@versant.com
Virgin Lands (Grafik, Cinematics)	Hauptstraße 5	97332 Volkach	09381 / 71780	info@virgin-lands.com
Walk Game Productions (Produktion)	Karl-Leopold-Straße 6	76229 Karlsruhe	0721 / 48399963	info@wolfgangwalk.de
Xaitment (KI-Middleware)	Fischbacherstraße 92	66287 Quierschied	06897 / 600800	info@xaitment.com
Z-Ground Illustration (Grafik)	Oeltzenstraße 17	30169 Hannover	0511 / 1696959	jz@z-ground.com

AUSBILDER

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
Akademie für Medien	Balinger Straße 39A	70567 Stuttgart	0711 / 42094444	h.ehlers@amk-net.de
Design Schule Schwerin	Bergstraße 40	19055 Schwerin	0385 / 5559775	info@designschule-schwerin.de
FH Trier	Schneidershof	54293 Trier	0651 / 8103345	sek@informatik.fh-trier.de
Games Academy Berlin	Rungestrasse 20	10179 Berlin	030 / 29779120	info@games-academy.de
Games Academy Frankfurt	Hanauer Landstraße 146	60314 Frankfurt	069 / 42696460	info@games-academy.de
Macromedia Hochschule für Medien und Kommunikation	Gollierstraße 4	80339 München	089 / 5441510	info.muc@macromedia.de
Mediadesign Hochschule für Design und Informatik	Lindenstraße 20-25	10969 Berlin	030 / 3992660	info-ber@mediadesign-fh.de
Mediadesign Hochschule für Design und Informatik	Werdener Straße 4	40227 Düsseldorf	0211 / 1793930	info-dus@mediadesign-fh.de
Mediadesign Hochschule für Design und Informatik	Berg-am-Laim-Straße 47	81673 München	089 / 4506050	info-muc@mediadesign-fh.de
Qantm Institute	Hoferstraße 3	81737 München	089 / 89068771	info@qantm.de
Qantm Institute	Soltauer Strasse 18-22	13509 Berlin	030 / 49860054	t.brueggemann@quantm.de
Qantm Institute	Linke Wienzeile 130A	A-1060 Wien	+43 / 1 / 961 03 03	andreas@sae.at
SRH Fachschulen	Bonhoefferstraße 1	69123 Heidelberg	06221 / 884488	info@fachschulen.srh.de
SRH Hochschule Heidelberg	Ludwig-Guttmann-Straße 6	69123 Heidelberg	06221 / 882790	tobias.breiner@fh-heidelberg.de
Zürcher Hochschule der Künste (ZHdK)	Ausstellungsstraße 60	CH-8005 Zürich	+41 43 / 4464646	info.ddk@zhdk.ch

VERBÄNDE

Firma	Straße/Hausnummer	Postleitzahl/Ort	Telefon	E-Mail
BIU	Rungestraße 18	10179 Berlin	030 / 24087790	office@biu-online.de
Cluster Audiovisuelle Medien Bayern	Sonnenstraße 21	80331 München	089 / 54460240	info@cam-bayern.de
G.A.M.E.	Frintroper Straße 453	45359 Essen	0201 / 613 17 21	reichart@game-bundesverband.de
Gamecity:Hamburg	Habichtstraße 41	22305 Hamburg	040 / 2270190	info@gamecity-hamburg.de
GAMEplaces	Hanauer Landstrasse 126-128	60314 Frankfurt am Mai		
Hessen-IT	Abraham-Lincoln-Straße 38-42	65189 Wiesbaden	0611 / 7748481	info@hessen-it.de
Mitteldeutsche Medienförderung	Hainstraße 17–19	04109 Leipzig	0341 / 269870	info@mdm-online.de
nordmedia	Expo-Plaza 1	30539 Hannover	0511 / 1234560	info@nordmedia.de
Northstar Developers	Bödekerstraße 84	30161 Hannover	0160 / 97880272	d.bruenig@collision-studios.com
ÖVUS	Aichhorngasse 14	A-1120 Wien	+43 / 1 / 810328912	office@ovus.at

Ein Tag bei ...
Radon Labs

Die Büros von Radon Labs liegen im Zentrum von Berlin direkt am Alexanderplatz.

Floral und anmutig präsentiert sich unser Empfangsbereich.

Das dynamische Duo im Office: Controllerin Izabela Gawlicka und Team-Assistentin Susann Krautwald..

Unser Talentscout: HR Managerin Doreen Kapahnke.

Game Designer Gustav Ziolkowski, Animator Robert Zwirner und Grafiker Florian Schambach im Gespräch – das Triumvirat grübelt über Wii-Spieleneuheiten.

Zwei unserer Drakensang-Autoren vor dem »Bildungszentrum« Aventuriens.

Ohne ihn geht bei Servern und Hardware nichts – System-Administrator Karl-Heinz Haag schraubt an neuen Rechnern.

Zwischen Blumen und Plüsch zeichnet Character Art Director Marcus Koch auf seinem Tablet.

Meeting in der Chill-Lounge: Hier beraten sich gerade Projektmanagerin Barbara Prodanovic, Executive Producer Sascha Elias und Lead Programmer Georg Leidinger.

Sammelt im Büro Pfand für die Berliner Tafel: unser Tester Timo Welak.

Die Drakensang-Entwickler sind umgezogen und präsentieren stolz ihre neuen Büros am Alex, die noch (!) gut gegossenen Zimmerpflanzen, ihren persönlichen Knut und selbstverständlich auch die Mega-Aussicht.

Oft sind die einfachsten Mittel die besten – klare Ansage auf dem Weg zum Arbeitsplatz.

Hier kommt Bewegung ins Spiel: unsere Character-Grafiker sitzen direkt neben den Animationsspezialisten.

Heiße Diskussionen: Flo & Flo bei einem der alltäglichen Meetings.

Programmiererin Susanne Schulz hat unschlagbare Argumente.

In der Küche trifft man sich zum Plausch und tankt beim Mittagessen neue Energien für kreative Ideen.

Ausgleichstraining Blumengießen: Technical Game Designer Christian Loch entspannt abseits der Excel-Listen.

Kein Berlin ohne (Eis-)Bär!

Die Chill-Lounge – feierabendlicher Treffpunkt der Wii-Girls von Radon Labs.

Ausgleich für Überstunden: der abendliche Blick aus dem Fenster!

In der nächsten Ausgabe
Making Games Vorschau

Making Games 01/2010

erscheint am 18. Dezember 2009

Titelthema:

Spieleförderung in Deutschland

Ob Gamecity Hamburg, Mitteldeutsche Medienförderung oder FilmFernsehFonds Bayern: Inzwischen gibt es Dutzende Förderinitiativen für Spiele-Entwicklungen. Wir wollen Ihnen in der kommenden Ausgabe die wichtigsten Programme vorstellen, sprechen mit den Entscheidungsträgern und geben Tipps für eine optimale Projektpräsentation.

Weitere Themen:

- **Interface Entwicklung**
  Lösungswege und Tools für eine reibungslose Zusammenarbeit von Design und Technik.
- **Zehn goldene Pitching-Regeln**
  Wir erläutern, auf welche Projektkriterien die Publisher derzeit besonders genau achten.
- **Ausbildung**
  Lügen und Wahrheiten über Praktika in der Games-Branche – ein Erfahrungsbericht.

Nutzen Sie uns!

Wir bieten Ihnen die Plattform für eine konstruktive Diskussion zum Standort Deutschland, nutzen Sie uns! Meinungsäußerungen und Vorschläge für Beiträge nehmen wir jederzeit entgegen. Schreiben Sie uns an **info@makinggames.de**.

IMPRESSUM

Projektleitung
Heiko Klinge (verantwortlich, Anschrift d. Red.)

Chefredaktion
Markus Schwerdtel

Chefredaktion
Christian Schmidt

Redaktion
Hendrik Weins

Redaktion
Petra Schmitz

Redaktion
Daniel Visarius

Redaktion (Praktikantin)
Sara Geller

Layout (ltd.)
Sigrun Rüb

Layout
Konstantin Christinakis

Director of Online & New Media Business
Gunnar Lott

Chefredaktion
Michael Trier

Redaktion
Michael Graf

Redaktion
Florian Klein

Redaktion
Daniel Matschijewsky

Redaktion (Trainee)
Yassin Chachchoukh

Redaktion (Trainee)
Thomas Wittulski

Lektorat
Marion Schneider

Layout
Eva Zechmeister

Freie Mitarbeiter dieser Ausgabe (Redaktion):
Björn Blankenheim, Stephan Krohn, Daniel Matzke, Kurt van Meter, Dominik Mieth, Chris Noeth, Claas Oehler, David Salz, Roman Pfrengle, Martin de Ronde, Wolfgang Siebert, Andreas Stock, Marc Ueberall, Pete Walentin, Philipp Wollmann, James Wyld

Wir danken den Interview-Partnern:
Florian Bohn, Dan Connors, Hendrik Klindworth, Sven Liebich, Elisabeth Mendoza, Simon Seefeld

Titelgrafik: h2design.de

Assistenz:
Isa Stamp, Anita Thiel, Annie Weissenberger

Korrespondenten:
Stacy Cowley (New York), Zhang Ran (Shanghai), Martyn Williams (Tokio), Peter Sayer (Paris), Sumner Lemon (Taipeh), George Dragon (Budapest), Lukas Erben (Prag)

Kundenservice

Abonnement und Einzelbestellung:
Making Games Kundenservice
ZENIT Pressevertrieb GmbH
Postfach 81 05 80, 70522 Stuttgart
Tel.: 0711 / 7252-278, Fax: 0711 / 7252-377
Österreich: Tel.: 01 / 219 55 60;
Schweiz: Tel.: 071 / 314 06-15
E-Mail: shop@makinggames.de
Web: www.makinggames.de/shop

Bezugspreise:
Einzelpreis: 6,90 Euro
Jahresabo: 28,98 Euro
(beinhaltet 6 Ausgaben)
Auslandspreise u. Mengenrabatt auf Anfrage

Zahlungsmöglichkeit:
Postbank Stuttgart, BLZ 600 100 70
Konto-Nr.: 311 704

So erreichen Sie die Redaktion:
IDG Entertainment Media GmbH
Redaktion Making Games
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-660, Fax: 089 / 360 86-652

Anzeigenkontakt

Anzeigenleiter:
Dirk Heib (-730), dheib@gamestar.de

Sales Manager Making Games:
Nicole Müller, 089 / 120 353 23, nmueller@idg-consultant.de

Stellv. Anzeigenleitung (Print):
Klaus Maurer (-673), kmaurer@gamestar.de

Senior Account Manager:
Annika Gredelewski (-625), agradelewski@gamestar.de

Junior Account Manager:
Susanne Fütterer (-670)
sfuetterer@gamestar.de

Sales Service Manager:
Daniela Kretschek (-691) Fax: (-672)

Digitale Anzeigenannahme Print:
Manfred Aumaier (-602)
anzeigendispoprint@gamestar.de

Digitale Anzeigenannahme Online:
IDG Sites E-Mail Sammeladresse:
banner@idg.de

Anzeigenpreise:
Es gilt die Preisliste gültig ab 01.01.2009

Vertrieb

Gesamtvertriebsleitung:
Josef Kreitmair (-243)
Vertriebsassistenz:
Melanie Stahl (-738)
Vertriebsmarketing:
Matthias Weber (-154, ltd.), Manuela Eue (-156)

Verlag

IDG Entertainment Media GmbH
Lyonel-Feininger-Straße 26
80807 München
Tel.: 089 / 360 86-0, Fax: 089 / 360 86-501
www.idgmedia.de

Geschäftsführer: York von Heimburg
Verlagsleitung: André Horn
Veröffentlichung gemäß §8, Absatz 3 des Gesetzes über die Presse vom 08.10.1949: Alleiniger Gesellschafter der IDG Entertainment Media GmbH ist die IDG Communications Media AG, München, die 100% Tochter der International Data Group Inc., Boston, USA, ist.

Vorstand:
York von Heimburg, Keith Arnot, Pat Kenealy
Aufsichtsratsvorsitzender:
Patrick J. McGovern
Produktionsleitung: Jutta Eckebrecht
Druck und Beilagen:
Mayr Miesbach GmbH, Druckerei und Verlag 83714 Miesbach

Haftung & Co.

© Copyright IDG Entertainment Media GmbH
Haftung: Für die Richtigkeit von Veröffentlichungen können Redaktion und Verlag trotz Prüfung nicht haften. Die Veröffentlichungen in Making Games erfolgen ohne Berücksichtigung eines eventuellen Patentschutzes. Auch werden Warennamen ohne Gewährleistung einer freien Anwendung benutzt.

RTL INTERACTIVE

THE QUALITY OF GAMES IS WHAT COUNTS

WWW.**BLACKINC-GAMES**.COM | WWW.**RTLSPORTS-GAMES**.COM | WWW.**PLAYTAINMENT-GAMES**.COM

BLACK INC

RTL SPORTS

PLAYTAINMENT

© RTL Television 2009, vermarktet durch RTL interactive GmbH. „RTL Sports", „Black Inc." und „Playtainment" sind Label der RTL interactive GmbH.

AUSTRALIA´S GAME EDUCATOR OF THE YEAR !!

qantm INSTITUTE

qantm.com • qantm.de • sae.edu

NL • GER • AU • UK • SG • AT • CH

Ausbildungsinstitut für Gamedesign & Development, Animation und kreative Medien

THE CREATIVE GAME & 3D EXPERTS

DIPLOMA & BACHELOR*
OF INTERACTIVE ENTERTAINMENT

KURSSTART:
März 2010 &
September 2010

INFOTAG:
29. November 2009
14:00 - 18:00 UHR

www.qantm.de • www.qantm.tv • www.sae.edu

Mail: berlin@qantm.de • muenchen@qantm.de

Berlin: +49 (0)30 43094470 - München: +49 (0)89 89068771

Zeichnung von Melanie Gwosdz IED0909

*in Kooperation mit der Middlesex University of London

sae INSTITUTE